第三届国家期刊奖百种重点期刊 | 中国科技核心期刊
工信部优秀科技期刊 | 中国五大文献数据库收录期刊

ISSN 1009-6868
CN 34-1228/TN

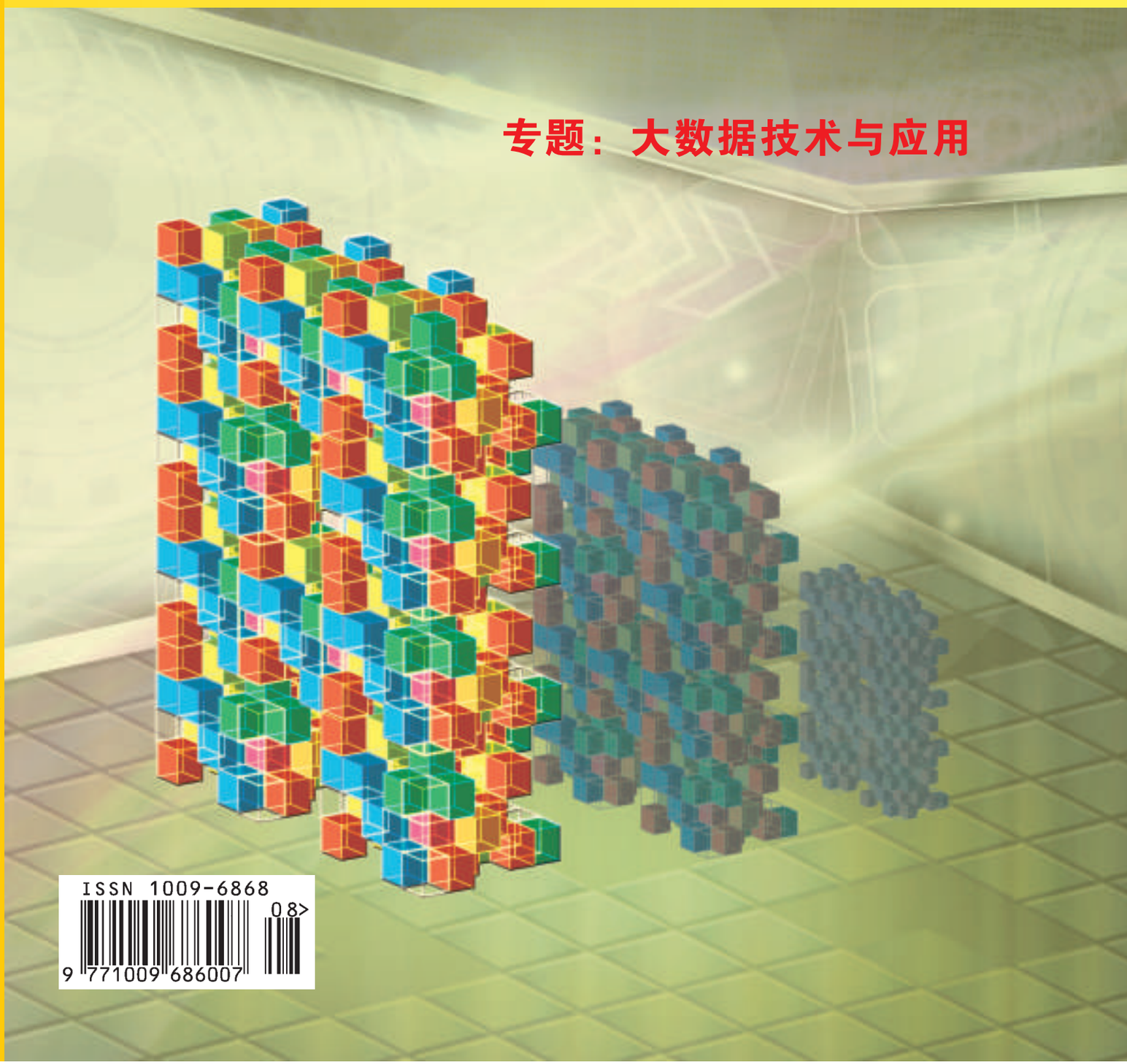
中兴通讯技术

ZTE TECHNOLOGY JOURNAL

www.zte.com.cn/magazine

2013年8月 • 第4期

专题：大数据技术与应用



ISSN 1009-6868
9 771009 686007 08>

中兴通讯技术 ZTE TECHNOLOGY JOURNAL

二〇一三年八月 第十九卷第四期 总第一一一期

目次

中兴通讯技术　总第111期　第19卷　第4期　2013年8月

专题：大数据技术与应用

专家视点

运营应用

研究论文

开发园地

系列讲座

综合信息

办刊宗旨

以人为本，荟萃通信技术领域精英；
迎接挑战，把握世界通信技术动态；
立即行动，求解通信发展疑难课题；
励精图治，促进民族信息产业崛起。

期刊基本参数：CN 34-1228/TN * 1995 * b * 16 * 64 * zh * P * ¥20.00 * 15000 * 13 * 2013-08

Contents

ZTE TECHNOLOGY JOURNAL Vol.19 No.4 Aug. 2013

Special Topic: Big Data and Its Applications

Expert View

Operational Application

Research Paper

Development Field

Lecture Series

敬告读者

邮购须知

本刊常年办理邮购订阅业务，欢迎订阅。订阅方法：从邮局汇款至编辑部，在汇款单上将订阅者的详细地址、收件人姓名及联系电话填写清楚，并在汇款单附言栏注明所购杂志期次及数量。

专题：大数据技术与应用

专|题|导|读

根据IDC的统计，每过18个月，人类所积累的数据总量就会增加1倍。全球在2010年正式进入ZB(zetabyte)时代；预计到2015年，每个联网用户每天将会生成超过4 GB的数据流量（相当于一部片长4小时的高清电影的流量）。物联网的蓬勃发展，更让数以亿计的各种传感器24小时不间断地采集海量数据。暴涨的数据量给传统计算和存储带来了前所未有的挑战，从而催生了云计算技术。

数据量暴涨的这一现象，其实用“海量数据”描述较为恰当，但这个词过于普通，已无法引起公众的兴趣。于是，有人便使用了“大数据”这个词。大数据的“大”和“云计算”的“云”异曲同工，从一开始就充分引起了公众的疑惑和好奇。慢慢地，人们感觉没有其他的词能够代替它们，因而使得大数据成为继云计算之后又一个在IT界炙手可热的名词。

在大数据相关技术的发展史中，Google公司有着举足轻重的地位。Google从2003年起，先后通过论文公开了Google文件系统(GFS)、并行分布式编程模型MapReduce、大数据数据库bigdata等技术，引起全球的效仿。2011年，Facebook效仿Google，将其全新设计的服务器和数据中心方案开源，在全球引起新型数据中心建设热潮。

本专题旨在揭示大数据现象所带来的历史性变革，阐述大数据应用的技术体系、关键技术与潜在问题，并介绍在国际处于领先水平的PB级超低功耗云存储系统和EB级实时云计算数据库，最后还探讨了基于云计算的大数据挖掘平台的构建，大数据在电信行业的解决方案及应用实践，以及大数据在面向城市信息感知的社交网络的应用案例。每篇论文都凝聚着作者对大数据研究的心血和汗水，希望这些成果与观点可以让读者受到启发，拓宽思路，开拓视野。在此，对各位作者的积极支持和辛勤工作表示衷心的感谢！

刘鹏

2013年5月20日

本期专题策划人

刘鹏

清华大学博士毕业；解放军理工大学教授、博导、学科带头人，中国云计算专家咨询委员会副主任/秘书长，中国电子学会云计算专家委员会云存储组组长；主要研究方向是信息网格、云计算；已主持完成基金项目18项；已发表论文80余篇，出版专著12部。

2013年第1—6期专题计划

1 自组织网络技术与应用
陈前斌 重庆邮电大学通信与信息工程学院院长

2 下一代互联网与IPv6技术演进
崔勇 清华大学计算机系网络所副所长

3 单波长太比特以上超高速光通信系统技术与器件
张成良 中国电信北京研究院副总工

4 大数据技术与应用
刘鹏 解放军理工大学教授

5 软件定义网络
王文东 北京邮电大学网络技术研究院副院长

6 移动互联网的发展趋势和技术方向
蒋林涛 工信部电信研究院科技委主任

DOI: 10.3969/j.issn.1009-6868.2013.04.001
网络出版地址：http://www.cnki.net/kcms/detail/34.1228.TN.20130624.1520.003.html

大数据——正在发生的深刻变革

Big Data: Profound Changes Taking Place

中图分类号：TN915.03; TP393.03 文献标志码：A 文章编号：1009-6868 (2013) 04-0002-006

刘鹏/LIU Peng
吴兆峰/WU Zhaofeng
胡谷雨/HU Guyu
（解放军理工大学 指挥信息系统学院，江苏 南京，210007）
(College of Command Information Systems, PLA University of Science and Technology, Nanjing 210007, China)

摘要： 介绍和比较了大数据在存储、管理、处理及挖掘方面全球主要的技术。大数据技术总的趋势是通过分布式计算来解决“瓶颈”问题。由于不能完全依赖提高单个节点性能的方式提升系统整体性能，因此需要通过增加系统内节点数目的方式来达到目的。可以将存储、处理和分析的任务通过分布式的方式分散到系统中各个节点上来加快数据的存储、处理和分析的速度。

关键词： 大数据；新摩尔定律；云计算；数据挖掘；Hadoop 平台

Abstract: In this paper, we describe and compare the main technologies for storing, managing, processing, and mining big data. Distributed computing is a new trend in solving bottlenecks associated with big-data development. Performance of the whole system cannot be improved only by improving the performance of a single node; therefore, it is necessary to increase the number of nodes within the system. Storage, processing and analysis can be distributed to each node in the system to speed up data storage, processing and analysis.

Keywords: big data; new Moore's law; cloud computing; data mining; Hadoop platform

随着人类对自然和社会认识地进一步加深及人类活动的进一步扩展，科学研究、互联网应用、电子商务、移动运营商等诸多应用领域产生了多种多样的数量巨大的数据。大数据（Big Data）的出现对传统的数据存储、数据处理及数据挖掘提出了新的挑战，同时也深刻地影响着人类的生活、工作及思维。传统的数据存储方法、关系数据库、数据处理和数据分析方法已不能满足当前的需要。

维基百科给出的大数据的定义如下：

巨量数据（或称大数据、海量资料），指的是所涉及的资料量规模巨大到无法透过目前主流软件工具，在合理时间内达到撷取、管理、处理，并整理成为帮助企业经营决策更积极目的资讯[1]。

目前工业界普遍认为大数据具有4V+1C的特征：

（1）数据量大（Volume）。存储的数据量巨大，拍字节级别是常态，因而对其分析的计算量也大。

（2）多样（Variety）。数据的来源及格式多样，数据格式除了传统的格式化数据外，还包括半结构化或非结构化数据，比如用户上传的音频和视频内容，而随着人类的活动的进一步拓宽，数据的来源更加多样。

（3）快速（Velocity）。数据增长速度快，同时要求对数据的处理速度也要快，以便能够从数据中及时地提取知识，发现价值。

（4）价值密度低（Value）。需要对大量的数据处理挖掘其潜在的价值，因而，大数据对我们提出的明确要求是设计一种在成本可接受的条件下，通过快速采集、发现和分析从大量、多种类别的数据中提取价值的体系架构。

（5）复杂度（Complexity）。对数据的处理和分析难度大。

1 大数据时代的来临

因特尔创始人戈登·摩尔（Gordon Moore）在1965年提出了著名的“摩尔定律”：即当价格不变时，集成电路上可容纳的晶体管数目，约每隔18个月便会增加1倍，性能也将提升1倍。1998年图灵奖获得者杰姆·格雷（Jim Gray）提出著名的“新摩尔定律”：每18个月全球新增信息量是计算机有史以来全部信息量的总和。我们可以将新摩尔定律同1439年前后古登堡发明印刷机时造成的信息爆炸作对比：在1453—1503年这50年间大约印刷了800万本书籍，比1200年之前君士坦丁堡建立以来整个欧洲所有手抄书还要多，即50年内欧洲的信息增长了1倍[2]；而现在的数据增长速度则是每18个月全球

收稿日期：2013-04-15
网络出版时间：2013-06-24
基金项目：国家科技重大专项（2012ZX03002003）

信息总量翻一番。图 1 可以清楚地看到大数据的增长，图 2 是 IDC 公司对未来全球数据总量的预测，图 3 则表明了大数据正在日益成为人们关注的焦点。我们已经进入到大数据时代。

2 大数据产生的原因

大数据随着人类活动的进一步拓宽而出现，他给我们带来了机遇也带来了挑战。

2.1 数据采集方式的改变

自从计算机诞生以来，特别是近几十年因特网的发展，人类逐步进入了信息社会。信息化时代一个关键特征是自动化，包括数据产生的自动化、数据处理的自动化等等，把人从简单繁琐的任务中解脱出来，用以解决需要创新的问题。比如在精细农业中，我们需要收集植物生长环境的温度、湿度、病虫害信息，来对植物的生长进行精细的控制。因此我们在植物的生长环境中安装各种各样的传感器，自动地收集我们需要的信息。自动化的出现使人类不再满足于得到部分信息，而是倾向于收集对象的全面的信息，即将我们周围的一切数据化（注意，这里并非“数字化”）。因此，美国提出了“数字地球”计划，因为在信息时代，谁掌握了信息的制高点，谁就能掌握主动权。而且有些数据如果丢失了哪怕很小一部分，都有可能得出错误的结论，比如通过分析人的基因组判断某人可能患有某种疾病，即使丢失一小块基因片段，都有可能导致错误的结论。这些原因都导致了我们必将面临数据的大爆炸。

2.2 人类活动范围的拓宽

在 Web2.0 时代，每个人不仅是信息的接受者，同时也是信息的产生者。全球每秒中发送 290 万封电子邮件，每天会有 2.88 万个小时的视频上传到 Youtube，Twitter 上每天发布的信息超过 5 千万条，每天亚马逊将产生 6.3 百万笔订单。2012 年 11 月 11 日 0 点起，天猫和淘宝网在 5 分钟内就有 1 000 万网民加入了抢购。

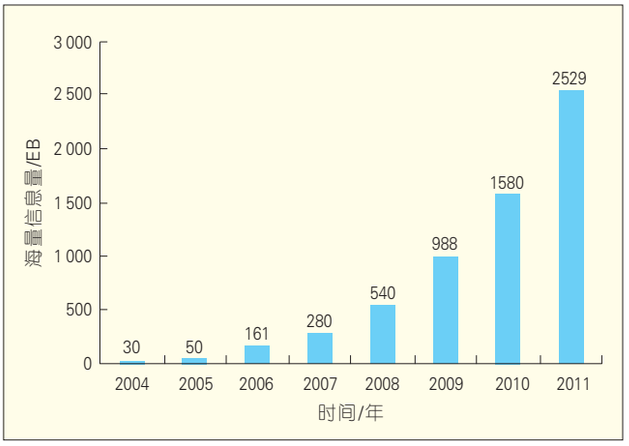


图 1 全球数据总量的增长

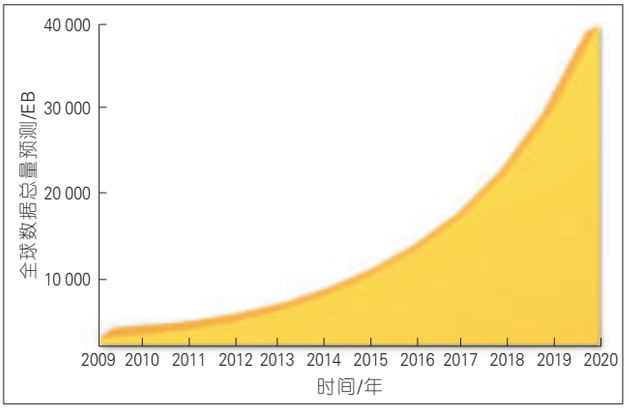


图 2 未来全球数据总量的预测

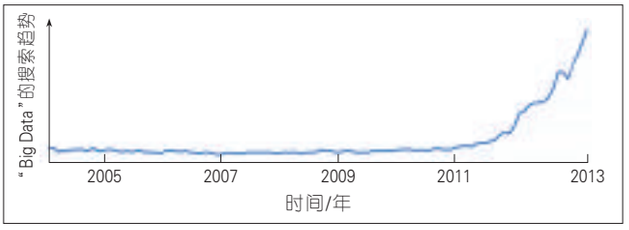


图 3 谷歌趋势反映的“Big Data”关键词的搜索趋势

以上只是我们见到的大数据的冰山一角，大数据的产生主要有下面几个来源：

（1）科学研究（包括天文学、生物学和高能物理等）。以天文学为例，2000 年斯隆数字巡天项目启动的时候，位于墨西哥州的望远镜在短短几周内收集到的数据比天文学历史上总共收集的数据还要多。

（2）智慧城市建设。智慧城市包括市政交通管理、精细农业、智能家居和环保监察等，以市政交通管理为例，在城市的任何道路上都可以看到摄像头，而视频数据是一个巨大的数据源。

（3）移动、电信和联通等通信和互联网运营商。运营商会实时采集网络底层数据进行网络优化，也会对所有用户的消费行为进行深度挖掘以制订相对应的营销策略。

（4）互联网企业（包括 SNS、微博、视频网站、电子商务）。这些最早接触大数据的企业，谷歌公司每天要处理 24 PB 大小的数据，中国所熟知

的互联网巨擘如百度、新浪、腾讯、阿里巴巴，每天产生的数据以拍字节量级计算。

2.3 大数据蕴含的潜在价值

从数据中发现知识，用以指导企业或者个人对生产和生活中碰到的问题进行决策，而不仅仅是产生报表。这些复杂的分析必须依赖复杂的分析模型，很难用结构化查询语言（SQL）语句进行表达，因此这类分析被称为“深度分析”。

以往的数据只是用来描述事实，进而理解产生这些数据背后的原因，现在我们需要通过对累积的数据进行分析，用以预测事物将来的发展趋势，进而采取相关的行动。在商业活动中，公司能够积累大量的交易记录，公司希望通过分析这些交易记录，找出其背后潜在的盈利模式。而SQL语句仅仅能够做到数据的呈现，无法满足找寻数据背后的相关性需求，进而探究事物之间的因果关系。谷歌在2009年初通过用户在网上的搜索记录成功预测甲型H1N1流感的爆发[3]。如果我们能够在流感爆发之前采取措施，将会给社会带来巨大的福祉。谷歌的成功预测是建立在大量数据的基础上。这就是大量数据背后的潜在的价值，谁能利用这些数据进行创新，谁就能够对未来的有更大的把握。为了得到数据背后的潜在价值，我们通常使用神经网络、数据挖掘及机器学习的方法建立模型，找出事物之间的关联，进而探究数据背后的原因，而这是单纯的SQL语句所无法胜任的。我们已经进入对大数据进行复杂分析的时代。

3 大数据解决方案

大数据时代的到来对数据的存储、处理及分析提出了新的挑战，但总的发展趋势是通过分布式计算来解决“瓶颈”问题。我们不能依赖提高单个节点性能这种纵向扩展的方式提升系统整体的性能，相反，我们需要能够通过增加系统内节点的数目这种横向扩展的方式来达到我们的目的。我们将存储、处理和分析的任务通过分布式的方式分散到系统中各个节点上来加快数据的存储、处理和分析的速度。

在实际的实现上，Google[4]、Amazon[5]、微软[6]和VMware[7]这4家公司在不同时间陆续推出各自的大数据方案，在应用领域和赢利模式上，Amazon和Google处于领跑者地位，微软和VMware紧随其后，此外还有开源的Hadoop[8]平台。Hadoop是谷歌大数据平台的开源实现，由于其开源特性，越来越多的企业在Hadoop的基础上对其进行修改以适应自己的需要，如Facebook根据其业务需求，底层采用Hadoop平台进行数据的存储和处理，并在其上开发了Hive[9]。Facebook通过Hive实现了例行性报表、即席查询、机器学习和数据挖掘算法，达到了较好的效果。图4是谷歌趋势描述的“Hadoop”关键词的搜索趋势。下面的对各项技术的比较过程中，我们将主要围绕这5种大数据解决方案展开比较。

3.1 大数据的存储

稳定、高效的存储系统既是系统正常运行的重要保证，也可以单独作为一项服务提供给用户。5种方案之中，Amazon的S3和微软的Blob存储比较类似，Google的GFS则完全不同，VMware目前仅向虚拟机提供存储服务，Hadoop仿照GFS开发了HDFS，是GFS的简化版本。相比GFS，HDFS缺少了多客户端并发的Append模型及快照功能。表1是5种存储服务的简单对比。

5种方案都提供了数据库存储服务。Google App Engine的Datastore构建在Bigtable上，但自身及其内部没有实现直接访问Bigtable的机制，可以看做是Bigtable上的一个简单接口。由于雅虎和Facebook的推动，Hadoop平台的数据库功能也越来越完善。雅虎在Hadoop平台上开发了Pig[10]，Facebook在Hadoop平台上开发了Hive，两者都是构建在HDFS之上，直接对HDFS进行操作，严格来讲只能算是对HDFS进行操作的接口。Hive目前已经开放了HBase接口，能够通过HBase对数据进行操作，因此，Hive同HBase的融合是未来发展的趋势。Amazon的SimpleDB采用的是“键/值”存储方式，功能比较简单，实现的查询功能也不太全面。SimpleDB和Datastore使用的都是“实体-属性-值”（Entity-Attribute-Value）的EAV数据模型。微软的SQL Azure是云环境下的关系数据库，并支持报表、数据同步等服务。10gen开发的开源云数据库MongoDB，可以实现均衡性较好的分布式数据库存储。Cassandra是Facebook推出的兼具Amazon Dynamo完全分布式特性和Google集中式管理特性的大数据库。数据立方是云创存储推出的列式完全分布式万亿记录级别的实时云计算数据库，其性能较之传统的云计算数据库提升约2个数量级[11]。表2是5种数据库之间的比较。

MapReduce[12]是谷歌提出的面向大数据的并行处理模型，具有扩展性好，鲁棒性高的优势，而属于关系型数据库的并行数据库是数据库发展的结晶，查询效率高，并且支持Schema。此外并行数据库的外围工

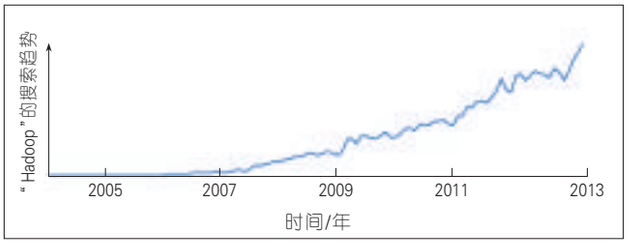


图4 谷歌趋势中反映的“Hadoop”关键词的搜索趋势

▼表1 大数据方案的存储服务比较

性能	Google GFS	Amazon S3	微软 Blob	VMware 存储	Hadoop HDFS
系统结构	文件分块存储	桶、对象两级模式	容器、Blob 两级模式	目录、文件两级模式	文件分块存储
可扩展性	可通过增加数据块服务器数量扩展存储容量	可通过增加桶中对象数量扩展存储容量	可通过增加容器中 Blob 数量扩展存储容量	自动迁移虚拟机以获取更大存储容量，及自动回收未使用存储容量	可通过增加数据节点数量扩展存储容量
数据交互方式	用户和数据块服务器进行数据交互	用户可以从获得授权的对象中取得数据	用户可以从获得授权的 Blob 中取得数据	仅提供给虚拟机使用	用户需要同名字服务器和数据服务器进行交互
存储限制	无特殊限制	桶的数量和对象大小有限制，但对象的数量无限	Blob 大小有限制，但是容器和 Blob 数量未限制	数据存储可跨越多个物理存储子系统	适合于大文件系统，小文件会削弱系统的性能
容量扩展方式	自动扩容	手动或编程实现自动扩容	手动或编程实现自动扩容	自动迁移虚拟机以扩容	自动扩容
容错技术	针对主、从服务器有各自的容错技术	数据监听回传 Merkle 哈希树数据冗余存储	仅重传出错的 Block，数据冗余存储	为运行中虚拟机创建与同步的 Shadow 虚拟机，多个虚拟机的集中备份	通过对名字服务器备份实现容错，此外通过对数据进行备份和校验进行纠错
负载均衡	由主服务器负责	弹性负载均衡	用户可以根据需要选用以下 3 种方法之一进行负载均衡：性能、故障转移、轮询	计算可能的迁移，同时考虑迁移的成本和效益，然后通过移动一个活多个虚拟机磁盘文件实现负载均衡	通过副本放置策略和负载均衡器来实现负载均衡

▼表2 大数据方案的数据库服务比较

性能	Google Datastore	Amazon SimpleDB	微软 SQL Azure	MongoDB	Hbase	Cassandra	数据立方
系统结构	实体组、实体、属性、值 4 级模式	域、条目、属性、值 4 级模式	Authority、容器、实体 3 级模式	集合、文档、域、值 4 级模式	行、时间戳、列簇	列、超级列、列簇、行、键值空间	表、记录、属性
主要存储的数据类型	结构化和半结构化数据	结构化数据	结构化数据	结构化和半结构化数据	结构化、半结构化数据	结构化、半结构化数据	结构化数据
所用的查询语言	GQL	支持有限的 SQL 语句	SQL	BSON	HiveQL、Pig Latin	专用 API	SQL、专用 API
数据更新时间	有延迟，但不是常态	有延迟	没有延迟	有延迟	不支持数据更新	不支持数据更新，但可以通过删除数据间接更新操作	没有延迟
实现的功能	较多	最少	最多	较多	较多	较多	较多
其他数据库服务	无	运行在 EC2 上的 Oracle、SQL Server 等	无	运行在 vCloud 上的 Oracle、SQL Server 等	无	无	运行于 cStor 云存储系统之上

具种类齐全，我们不能因为大数据就把这些非常好用的软件全部扔掉，这样做不经济也不合理，在小规模数据和数据的报表显示方面，这些工具性能卓越。目前越来越多的研究人员逐渐意识到，MapReduce 技术和并行数据库的融合才是真正的解决大数据问题的有效途径[13]。文献[14]指出，目前并行数据库同 MapReduce 的融合包括 3 个方面：

（1）并行数据库主导型，典型的代表有 Exadata、Greenplum 等。

（2）MapReduce 主导型，典型代表有 Hive 和 Pig。

（3）并行数据库和 MapReduce 集成型，典型代表有 HadoopDB、Vertica 及 Teradata 等。

3.2 大数据的处理

计算服务是所有的大数据解决方案最核心的业务之一，同时也是用户最常用的服务。Google 和 Hadoop 提供基于 MapReduce 的数据处理，整个过程对用户而言是透明的。Amazon 的 EC2 给予用户配置硬件参数的权利，使得用户可以根据实际的需求动态地改变配置，从而提高效率和节省资源。微软的 Azure 允许用户在处理数据之前设置部分参数，但相对于 EC2 其灵活性要差很多。VMware 的 vCloud 中提供了 DRS 和 DPM 技术，可以通过迁移和关闭虚拟机来实现资源优化。表 3 是这 5 种计算服务的比较。

MapReduce 在系统层面解决了大数据分析平台的扩展性和容错性问题，是非关系型数据库的典型代表，因此越来越多的研究人员从性能和易用性方面对 MapReduce 进行改进。对 MapReduce 性能提升的研究包括 4 个方面：

（1）多核硬件与图形处理器上的性能改进。

（2）索引技术与连接技术的优化。

（3）调度技术优化。

（4）其他优化技术。

针对 MapReduce 易用性的研究成果包括 Yahoo 的 Pig、Microsoft 的 LINQ、Hive 等。

从上述比较中不难发现，5 种大数据解决方案在大数据的存储和处理方面都存在较大的差异。但不同方案之间没有绝对的优劣之分，仅有适用场合的区别，用户可在确定自身的需求后进行选择。

3.3 大数据的数据挖掘

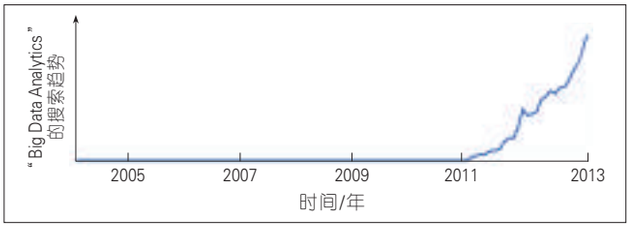
3.3.1 大数据数据挖掘的背景

图 5 是谷歌趋势反映的“Big Data Analytics”关键词的搜索趋势，可以看出对大数据的分析已经成为关注的焦点。对大数据进行分析，找出其背后的潜在关系，是大数据的最终目的，只有这样大数据才能真正释放其价值。

互联网和电子商务企业应用数据挖掘算法对大数据进行分析的做法由来已久。谷歌通过将软件免费提供给用户使用，使其能够对用户的喜好进行分析，从而定制更加具有针对性的广告策略。此外，谷歌在机器翻译和图像识别方面的成就也是有目共睹，而这一切都建立在对大量的数据进行分析的基础上。亚马逊能够根据以往用户的购买记录向用户

▼表3 大数据方案的计算服务比较

性能	Google MapReduce	Amazon EC2	微软 Azure 计算服务	VMware vCloud 计算服务	Hadoop MapReduce
服务类型	PaaS	IaaS	PaaS	IaaS	PasS
虚拟机的使用	未使用	用户可以根据需要设置运行虚拟机的硬件配置	系统自动分配	vCenter 自动进行资源优化	用户能够自行决定是否使用虚拟机
运行环境	Google 自身提供的环境，用户无法自行调配	用户自行提供运行程序所需的 AMI	系统自动为用户生成的装有 Windows Server 2008 的虚拟机	用户在虚拟机、虚拟设备和 vApp 3 种模式中选择一种	用户能够根据自己的需要进行配置
易用性	最好	稍差	较好	较好	最差
灵活性	稍差	最好	较好	较好	较好
适用的应用程序	适合可以并行处理的应用程序	任意程序	任意可在 Windows Server 2008 上运行的程序	任意程序	适合可以并行处理的应用程序



◀图5 谷歌趋势反映的"Big Data Analytics"关键词的搜索趋势

推荐相似的商品，这项技术为亚马逊带来了巨大的收益，作为消费者，我们也很难不受这些推荐内容的影响。现在我们已经能够通过数据挖掘预测飞机票、规划最佳线路及对汽车的安全状况进行监测等。这些都是对大数据进行挖掘的例子。通过数据挖掘，能够为公司带来巨大的利益，也能使我们的生活更加便利。

在中国，中国移动在2007年3月确定实施"大云"计划，并同中科院计算所合作开发了大云数据挖掘系统（BD-PDM）。该系统是一套高性能、低成本、高可靠性、高可伸缩性的海量数据处理、分析和挖掘系统，实现了数据的分类、聚类及关联规则发现。阿里巴巴利用Hadoop平台对海量电子商务交易数据进行存储和深度数据挖掘，并于2011年启动10亿元云基金，专注于基于云计算的电子商务、分布式存储和计算技术、数据中心运维技术、大规模/超大规模的数据挖掘和分析的算法等等。

Mahout[15]是一个基于Hadoop的开源数据挖掘平台，其主要目标是创建一些可伸缩的机器学习算法，供开发人员在Apache许可下免费使用。虽然其在开源领域比较年轻，但已经提供了大量功能，特别是在集群和CF方面。

3.3.2 大数据数据挖掘的研究现状

当数据规模增大的时候，已有的数据挖掘算法已经不再适用，需要对其改进，以利用并行计算模型加快数据的处理速度。目前对大数据进行数据挖掘的研究大致包括3个方面。

（1）集中在将已有的在单个机器上运行的机器挖掘算法迁移到并行计算平台上来。文献[16]提出了一种基于MapReduce的、适用于大量机器学习算法的通用并行编程框架，在该框架下，他们实现了包括线性回归、朴素贝叶斯等在内的10种经典的数据挖掘算法。文献[17]阐述了SVM在在MapReduce模型下的实现。文献[18]提出了Parallel FP-Growth算法，并通过实验证明了该算法具有极强的扩展性，适用于海量数据挖掘。文献[19-22]也都是对已有的数据挖掘算法进行改进，使其能够通过MapReduce并行计算模型加快计算速度，以适应大数据背景下的数据挖掘要求。

（2）利用MapReduce并行计算模型解决具体的问题。文献[23-24]都是对Web数据进行数据挖掘。文献[23]利用MapReduce模型改进并优化了Web数据挖掘中的Graph算法，文献[24]重新设计和实现了基于中文词网络的HITS算法，对该算法进行Map/Reduce化，并测试和分析了实验结果。文献[25]分析了中药的复方数据，发现了中药药物网络具有复杂网络特性，并采用MapReduce并行计算模型对分析复杂网络的算法进行了并行化处理。文献[26-28]也都是在具体的应用中通过MapReduce模型对已有的算法进行并行化处理。

（3）利用已有的数据挖掘算法构建大数据挖掘平台。通过将已有的数据挖掘算法同大数据挖掘平台的集成，能够使我们在利用已有的研究成果的同时，快速地开发相关的算法，使我们专注于实际的应用问题。已有的开源数据挖掘平台R和Weka被广泛使用。文献[29-30]致力于将R和Hadoop集成，使Hadoop获得强大的分析能力。文献[31]实现了Weka和MapReduce的集成。

4 结束语

大数据的产生是必然的，而且已经在深刻地影响着我们的工作和生活。本文分析了大数据的产生与发展，并对大数据的存储和处理及对大数据的数据挖掘作了介绍，最后对本文作了总结。我们有理由相信，在不远的将来，大数据将带给我们更多的精彩。我们应当抓住机遇，在未来出现的大数据生态系统中找到自己的一席之地。

参考文献
[1] 大数据 [EB/OL]. [2013-04-13]. http://zh.wikipedia.org/zh/Big_data.
[2] 迈耶-舍恩伯格, 库克耶. 大数据时代：生活、工作与思维的大变革 [M]. 盛杨燕, 周涛, 译. 杭州：浙江人民出版社, 2013:1-23.
[3] GINSBERG J, MOHEBBI M H, PATEL R S, et al. Detecting influenza epidemics using search engin query data [J]. Nature, 2009, 457 (19): 1012-1014.

[4] Google. Google App Engin [EB/OL]. [2013–04–13]. http://code.google.com/appengine/.
[5] Amazon. Amazon Web Service [EB/OL]. [2013–04–13]. http://aws.amazon.com/.
[6] Microsoft. Introducing the Windows Azure Platform (Final PDC10) [EB/OL]. [2013–04–13]. http://www.windowsazure.com/en–us/develop/net/fundamentals/intro–to–windows–azure/.
[7] VMware. VMware vCloud [EB/OL]. [2013–04–13]. http://www.vmware.com/products/vcloud/.
[8] Hadoop [EB/OL]. [2013–04–13]. http://hadoop.apache.org/.
[9] THUSOO A, SARMA J S, JAIN N, et al. Hive a warehousing solution over a MapReduce framework [J]. Proceedings of the VLDB Endowment (PVLDB), 2009, 2(2): 938–941.
[10] OLSON C, REED B, SRIVASTAVA U, et al. Pig Latin: A not–so–foreign language for data processing [C]//Proceedings of the 2008 ACM SIGMOD international conference on Management of data (SIGMOD'08), Jun 9–12, 2008,Vancouver, Canada.New York, NY, USA: ACM, 2008: 1099–1110.
[11] cStor [EB/OL]. [2013–04–13]. http://www.cstor.cn.
[12] DEAN J, GHEMAWAT S. MapReduce: Simplified data processing on large clusters [C]//Proceedings of the 6th USENIX Symposium on Operating System Design and Implementation (OSDI'04),Dec 6–8, 2004,San Francisco, CA,USA. Berkeley, CA, USA: USENIX Association, 2004:137–150.
[13] 覃雄派, 王会举, 杜小勇, 等. 大数据分析——RDBMS与MapReduce的竞争与共生 [J]. 软件学报, 2012, 23(1):32–45.
[14] 王珊, 王会举, 覃雄派, 等. 架构大数据:挑战、现状与展望 [J]. 计算机学报, 2012, 34(10): 1741–1752.
[15] Mahout [EB/OL]. [2013–04–13]. http://mahout. apache.org/.
[16] RANGER C, RAGHURAMAN R, PENMETSA A, et al. Evaluating MapReduce for multi–core and multiprocessor systems [C]// Proceedings of the IEEE 13th International Symposium on High Performance Computer Architecture(HPCA'07),Feb 10–14, 2007, Phoenix, AZ,USA. Piscataway, NJ, USA: IEEE, 2007:13–24.
[17] CHANG E Y,ZHU K H, WANG H, et al. PSVM: Parallelizing support vector machines on distributed computers [C]// Proceedings of the 21st Annual Conference on Neural Information Processing Systems (NIPS'07), Dec 3–6,2007,Vancouver, Canada. Berlin, Germany: Springer–Verlag, 2007:213–230.
[18] LI H Y, WANG Y, ZHANG D, et al. Parallel FP–growth for query recommendation [C]// Proceedings of the 2nd ACM Conference on Recommender systems (RecSys'08), Oct 23–25, 2008, Lausanne, Switzerland. New York, NY, USA: ACM, 2008:107–114.
[19] 郁洋. 基于云计算的并行聚类算法研究 [D]. 南京: 南京邮电大学, 2012.
[20] 陈爱平. 基于Hadoop的聚类算法并行化分析及应用研究 [D]. 西安: 西安电子科技大学, 2012.
[21] 张明辉. 基于Hadoop的数据挖掘算法的分析与研究 [D]. 昆明: 昆明理工大学, 2012.
[22] 李曼. 云计算平台上的增量学习研究 [D]. 南京: 南京邮电大学, 2012.
[23] 李雪锋. 基于云计算环境的web数据挖掘算法研究 [D]. 北京:北京交通大学, 2010.
[24] 李辉. 基于与计算环境的web结构挖掘算法研究 [D]. 杭州: 浙江理工大学, 2012.
[25] 刘正. 基于MapReduce的中药数据网络化及挖掘 [D]. 南京: 南京大学, 2012.
[26] 李彬. 基于MapReduce编程模型的航空日志分析研究 [D]. 成都: 成都理工大学, 2012.
[27] 高进. 基于MapReduce的DNA序列拼接算法研究 [D]. 北京: 北京交通大学, 2012.
[28] 肖韬. 基于MapReduce的信息检索相关算法并行化研究与实现 [D]. 南京: 南京大学, 2012.
[29] DAS S, SISMANIS Y, BEYER K S, et al. Ricardo: Integrating R and hadoop [C]// Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD'10),Jun 6–10, 2010, Indianapolis, IA, USA.New York, NY, USA: ACM,2010: 987–988.
[30] SAPTARSHI G, RYAN H, JEREMIAH R, et al. Large complex data: Divide and recombine(D&R) with RHIPE [J]. The ISI's Journal for Rapid Dissemination of Statistics Research, 2012, 1(1):53–67.
[31] WEGNER D, MOCK M, ADRANALE D, et al. Toolkit–based high–performance data mining of large data on MapReduce clusters [C]//Proceedings of the 9th IEEE International Conference on Data Mining (ICDM'09), Dec 6–9, 2009,Miami, FL, USA. Los Alamitos, CA, USA: IEEE Computer Society,2009: 296–301.

作者简介

刘鹏，清华大学博士毕业；解放军理工大学教授、博导、学科带头人，中国云计算专家咨询委员会副主任/秘书长，中国电子学会云计算专家委员会云存储组组长；研究方向为信息网格、云计算；已主持完成基金项目18项；已发表论文80余篇，出版专著12部。

胡谷雨，解放军理工大学首席教授、博士生导师，江苏省有突出贡献中青年专家，国家科技进步奖评审专家；研究方向为计算机网络、网络管理、网络智能；已发表学术论文160余篇。

吴兆峰，解放军理工大学在读博士研究生；研究方向为计算机网络。

综合信息

2013年1—5月中国信息消费规模同比增长19.8%

工业和信息化部运行监测协调局发布数据显示：2013年1—5月，全国信息消费规模不断扩大，网络和信息基础设施进一步完善，信息服务和应用创新活跃，居民消费潜力迅速扩大，电子商务增势迅猛，产品智能化应用日益突出。

数据显示：2013年1–5月中国信息消费规模1.38万亿，同比增长19.8%。其中通信业务收入4 658.8亿元，同比增长8.7%；软件技术服务消费4 590.4亿元，同比增长25.9%；信息终端产品消费4 019万元，同比增长25.7%。电子商务快速发展，累计交易额达40 150亿元，同比增长46%。

智能信息终端普及加快，贡献率提升。1–5月笔记本电脑、彩电和移动通信手持机的内销量分别为2 525、3 328和20 100万台。终端产品的智能化日益突出，1–5月智能手机销量同比增长110.8%，内销占比达76%；其中新机型的智能机占比达87.4%，比上月提高1.2%。

信息服务方式不断创新，居民消费习惯变化明显。据CNNIC调查显示：目前28.4%用户习惯使用网络获取社会消费品信息，手机购物应用进一步挖掘了消费者闲暇时购物的欲望和潜力，使用手机登录网站浏览的用户达53.6%。（转载自C114中国通信网）

DOI: 10.3969/j.issn.1009-6868.2013.04.002
网络出版地址：http://www.cnki.net/kcms/detail/34.1228.TN.20130627.1215.002.html

大数据应用的技术体系及潜在问题

Big Data: Technical Ecosystem and Problem Discovery

中图分类号：TN915.03; TP393.03　文献标志码：A　文章编号：1009-6868 (2013) 04-0008-009

窦万春/DOU Wanchun
江澄/JIANG Cheng
（南京大学 计算机科学与技术系，江苏 南京，210023）
(Department of Computer Science and Technology, Nanjing University, Nanjing 210023, China)

摘要： 大数据处理流程包括：数据获取、数据集成、数据分析和解释3个阶段。大数据应用的技术和系统包括：云计算及其编程模型MapReduce、大数据获取技术、面向大数据处理的文件系统、数据库系统、大数据分析技术。大数据应用所面临的问题包括：人力和财力问题、安全和隐私问题、生态环境和产业链的变革问题。

关键词： 大数据；云计算；MapReduce技术

Abstract: There are three steps in processing big data: data acquisition, data integration, data analysis and interpretation. In these steps, cloud computing, MapReduce, data acquisition techniques, data processing systems, database systems, and data analysis techniques may be used. In big-data applications, there are human and financial issues, security and privacy issues, environment and industrial chain issues, and transformation issues.

Key words: big data; cloud computing; MapReduce

随着智能手机等移动设备的普及以及无线网与Web2.0接口技术等方面的发展，网络用户的数量正迅猛增长，随之而来的是社交网络活动的日益频繁和数据服务需求的逐渐增多。据统计，2010年的手机用户已达40亿，占全球人口的60%，其中智能手机占了12%，用户年增长率达到了20%[1]。

众所周知，物联网近年来已成为普遍关注话题，实现物联网的宗旨就是让所有能被独立寻址的普通物理对象实现互联互通的网络，由此传感器与射频识别（RFID）等相关无线技术得到了迅速发展，调查显示，2011年已经有3 000多万的传感器节点遍布在交通运输业、工业以及零售业等场所，并且这个数值以每年30%的增长率提升中。而RFID由于其强大的无线传输和处理能力，也使得其遍布在各领域，用来实现清单管理的自动化[2]。这些传感器和RFID无时无刻不产生着大量的数据。具体地，谷歌在2008年的日均处理数据量已达20 PB；亚马逊在2010年11月29日这天的峰值交易数是158笔每秒；一架波音737飞机飞行6小时所产生的传感器数据达到240 TB[3]。IBM估计，每天由人类和机器产生的初始数据竟然达到了2.5×1 019字节[4]。这一切都为大数据时代的到来酝酿了潜在的应用需求。

面对大数据时代的到来，各国各组织都在积极着手准备应对策略。继Nature在2008年推出大数据专刊后[5]，2011年瑞士达沃斯世界经济论坛上，大数据成为重要主题，论坛中的一份“大数据，大影响”的报告指出了大数据如今已成为了像黄金和外汇一样的一种新型的经济资产。在美国，奥巴马政府于2012年3月公布了“大数据研究和发展的倡议”[6]，投资2亿多美元开启大数据研发计划；紧接着，中国在2012年5月召开的第424次香山科学会议，是中国第一个以大数据为主题的重大科学工作会议，随后中国计算机学会、通信学会也随即分别成立了大数据专家委员会；2013初，澳大利亚政府也在堪培拉的信息行业协会峰会上表示，将于5月出台大数据战略草案。上述学术与社会活动表明，大数据已然成为了学术界和工业界等各界关注的重要课题，并且已经悄然影响到当今人们的日常生活。

大数据时代的到来，挑战与机遇并存。当传统关系数据库管理技术由于自身的扩展性限制，已无法继续很好地适用于大数据处理的时候，云计算应运而生，并迅速成为热门话题，2004年谷歌提出的MapReduce作为面向大数据处理的计算模型[7]，更是倍受学术界和工业界的青睐。为此，本文首先对大数据的基本概念进行了阐述，讨论了大数据处理的流程、云计算和MapReduce等相关技术，然后分析了大数据带来的问题，

收稿日期：2013-04-19
网络出版时间：2013-06-27
基金项目：国家科技重大专项（2011BAK21B06）；国家自然科学基金（61073032）

最后总结全文并对大数据处理进行了展望。

1 大数据概述

1.1 大数据的定义

维基百科对大数据的定义是，所涉及的资料量的规模巨大到无法透过目前主流软件工具，在合理时间内达到撷取、管理、处理、并整理成为帮助企业经营决策更积极目的的各种资讯。

大数据目前主流的对大数据的定义为3V，即规模性（Volume），多样性（Variety）和高速性（Velocity）。所谓规模性，就是数据的量达到了一定的高度，无法通过当前主流工具来及时处理；多样性指的是对于即将要处理的数据类型，除了有结构化的以外，还有半结构化和非结构化的，增加了操作的复杂性；高速性是指数据的到达与处理必须及时高效，不允许较长的延迟[8]。除此之外，一般也认为，隐私性与有价值型同样是大数据的主要特征[9]。

1.2 大数据的带来的机遇与挑战

随着大数据时代的到来，其中隐藏的商机也被各路商家发现和利用。美国Target百货公司通过一套客户分析工具，可以对顾客的购买记录进行分析，并随后通过购物手册的形式向顾客推荐一系列可能需要的商品；“阿里云”通过对其云平台上海量的交易和数据进行分析，从而知道哪些商户可能存在资金问题，随后“阿里云”贷款平台便出马同潜在的贷款对象进行沟通；“京东”、“天猫”和“易购”等购物网站将其海量商品按照各种方式进行分类和推荐，大大增强了网站的可用性。

国际著名的市场调研公司“高德纳”公司的一份分析报告指出，到2015年，使用先进数据管理系统的企业将比未使用的企业盈利能力高出20%。咨询公司“益百利”集团的研究也表明，2012年全球对大数据项目的投资总额大约达45亿欧元，预计后两个年度均将保持大约40%的增长速度。

不单是商家，大数据处理技术也给普通用户的日常生活带来了方便性和可靠性。购物网站可以使用户足不出户便可购买到廉价优质的商品，地图软件让人们出门再也不用担心迷路的问题，“微信”、“微博”使得人们随时随地能够跟亲人、朋友联络交流，各种互动娱乐软件帮助人们打发无聊地时光等等。

1.3 大数据处理流程

大数据带来的利益不可小觑，由于大数据的规模性、高速性、多样性等本质决定了其处理过程的复杂性，而如何处理大数据却成为一道难题摆在了人们面前。图1所示为大数据处理的一般流程。

大数据处理流程一般可分为数据获取阶段、数据集成阶段以及数据分析解释阶段。

1.3.1 数据获取阶段

数据获取阶段主要是完成对外界数据源的接收和记录操作。其中对大数据的接收方式主要有传感器获取、网页点击获取、移动设备上应用服务的获取以及RFID获取等；对大数据的记录主要完成对元数据的选择，以便构建所需要的数据结构。

1.3.2 数据集成阶段

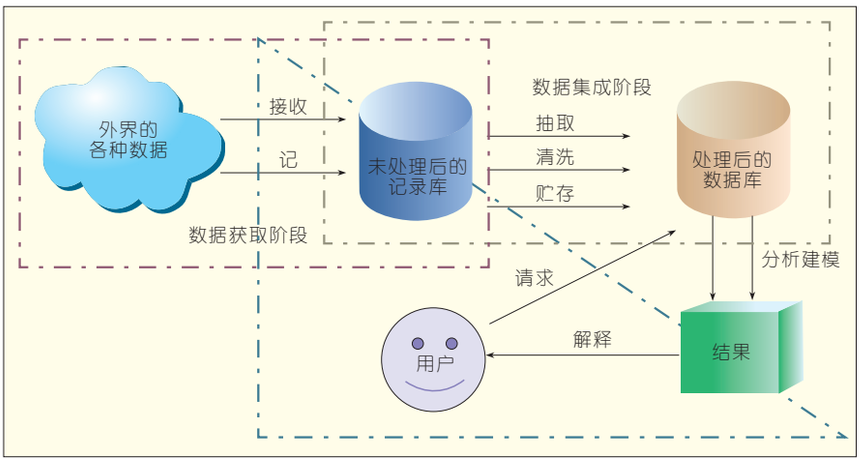
大数据的集成阶段主要完成对已接收数据的抽取、清洗和贮存等操作。

（1）抽取

由大数据的定义可知，获取的数据可能具有多种结构和类型，数据抽取过程可以帮助我们将这些复杂的数据转化为单一的或者便于处理的构型，以达到快速分析处理的目的。例如，对于一起交通事故的发生，目击者的口述表达与监控摄像头的拍摄显示虽然都是在一定程度上反映了事故的经过，但由于数据格式的不同，不方便对此类问题做大规模的统计分析，将这些数据统一转化为标准的表示格式将会大大地方便后期的分析工作。

（2）清洗

对于大数据，并不全是有价值的，有些数据并不是我们所关心的内容，而另一些数据则是完全错误的干扰项，如何“去噪”从而提取出有效数据对我们来说是个巨大挑战。其中一种做法是设计一些过滤器，通过某些规则将那些无用错误的数据过滤出去，防止对最后的分析工作产生影响。例如，对于交通事故的描述，有些目击者或者当事人出于某些主观或者客观原因，提供了一些模糊或者



▲图1 大数据处理流程

虚假的信息，对这些信息的过滤操作非常重要。

（3）贮存

将初步处理过得数据进行有效的存储至关重要，若是仅仅将这些记录随便地放入一个数据仓库中，将会造成其访问性受到障碍，从而可能导致了数据的难以复用。设计一个合适的数据库，可以有效地解决难以复用问题。

数据库的选择可以多种多样，针对特定数据设计的特定数据库将会更加高效、适用。

1.3.3 数据分析和解释阶段

当用户提出查询请求时，我们需要做的就是进行及时地分析与建模，并将结果以用户可接受的方式返回给用户。这一阶段的用户查询可以是多种多样的，不同的查询输入应该得到对应的结果，即使面对用户的错误查询也应该给出相应的错误友好处理。

分析、建模的过程多种多样，统计学、数据挖掘、机器学习等各类方法相互结合可以产生各种智能推荐系统以满足用户的查询请求。庞大的数据量虽然处理起来比较麻烦，但往往能让我们从中发现更有价值的信息。

当然，用户并不是专业的技术人员，如何将查询结果解释给用户至关重要。一个良好的系统，应该不仅仅告诉用户不同输入对应的不同结果，更要以通俗易懂的方式告知用户相应地结果是如何产生的，从而让用户有更可信的感觉。对于那些模糊甚至错误的查询请求，应该能够通过大数据的海量联系发掘并纠正这类请求，从而更加人性化。当然，大数据处理的及时性要求我们应当更快更及时的处理用户查询，决不允许较大的处理延迟。

总之，大数据的本质决定了大数据的分析处理具有复杂性与独特性，同时也带来了相对于普通数据处理所没有的可靠性与可用性。

2 大数据应用的技术体系

2.1 云计算及其编程模型 MapReduce

2.1.1 云计算简述

大约从2007年下半年开始，云计算由于其能提供灵活动态的IT平台，服务质量保证的计算环境以及可配置的软件服务而成为热门话题[10]。文献[11]中给出了云计算的比较完整的定义：云计算一个大规模的由规模经济驱动的分布式模型，位于其中的抽象的、虚拟的、动态可扩展的、可管理的计算能源、存储、平台、服务等通过因特网交付给外围客户。

由上述云计算的定义我们知道，云计算首先得是大规模的、分布式的，少量的计算处理用不着云计算；其次，它是跟规模经济相关联的，比较形象的说法是，云计算资源跟“电”和“水”一样，是按需收费的，并且是大规模式销售的，通常在建立数据中心时会考虑成本因素；最后，它从广义上说是给客户的一种服务，可以包括提供存储、计算等资源。云计算可以按服务的内容和交付形式分为基础设施即服务（IaaS）、平台即服务（PaaS）、软件即服务（SaaS）等。

在单机芯片集成度已进入极小尺度级别，指令级并行度提升也已接近极限的今天，纵向扩展似乎已经不够现实，这也远远不能满足大数据处理的要求，而云计算的要求比较宽松的允许异构网络的横向扩展，无疑给大数据处理带来了方便。云计算能为大数据提供强大的存储和计算能力，可以迅速、方便地为大数据提供服务，另一方面，大数据的处理需求也为云计算提供了更多更好地应用场景。由此，云计算作为大数据的支撑技术而倍受业界关注。

2.1.2 MapReduce 简述

关系数据库作为一门发展了近40年的主流数据管理技术，主要用于联机事务处理（OLTP）应用、联机分析处理（OLAP）应用和数据仓库等，然而扩展性方面的局限使得其在大数据时代遇到了极大障碍。2004年，谷歌公司提出的MapReduce技术，以其利用大规模廉价服务器以达到并行处理大数据的目的而倍受学术界和工业界的关注，广泛应用于机器学习、数据挖掘等诸多领域。基于MapReduce的大数据分析处理研究也在不断深入，MapReduce作为一种非关系数据库的数据管理工具代表，克服了关系数据库扩展性方面的不足，将计算推向数据也迎合了大数据时代的内在需要，成为大数据处理的基本工具。

Hadoop作为模仿谷歌公司提出的MapReduce而实现的一个云计算开源平台，目前已成为最为流行的大数据处理平台。

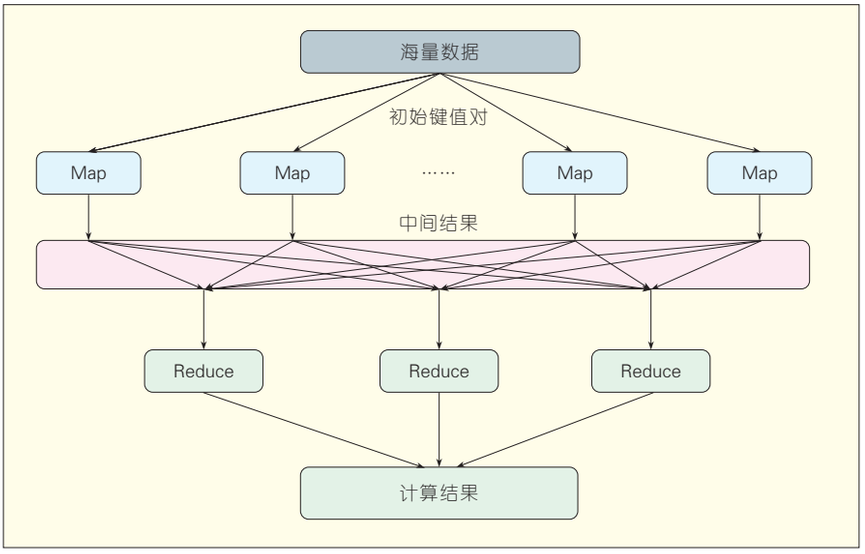
MapReduce对于大数据处理的基本构思是分而治之，将大数据任务分解为多个子任务，将得到的各个子结果组合并成为最终结果。

MapReduce对大数据的处理可抽象为两个主要阶段，Map阶段先对初始的键-值（Key/Value）对进行处理，产生一系列的中间结果Key/Value对，然后再通过Reduce阶段合并所有具有相同Key值的Key/Value对，得到最终结果。

MapReduce对数据进行处理的应用思路如图2所示。

MapReduce并行处理流程（待处理的大数据被分为大小相同的块）主要步骤为：

- 用户作业程序提交给主节点
- 主节点为作业程序寻找和配备可用的Map节点和Reduce节点
- 主节点启动Map节点执行程序，读取本地数据
- 每个Map节点处理读取的数据块，将中间结果放在本地并通知主节点计算完成及结果数据存储位置
- 主节点启动Reduce节点运行，



▲图2 MapReduce处理数据的基本思路图

远程读取中间结果并处理

2.2 大数据获取技术

每天都有大量数据产生，并且这些数据通过不同的途径，以不同的形式被接收和记录。本节将简单介绍几种常见的大数据获取途径。

（1）传感器技术

近年来，传感器技术蓬勃发展，无论是道路交通方面，还是医疗机构方面甚至是个人工作和生活场所，传感器无处不在，大量的数据源源不断地被传感器所接收。可以说，传感器的迅速普及，为大数据的获取提供了有力地保障。

传感器技术的快速发展，也促进了传感器网络的逐步完善。由于构建传感器网络的设备、数据收集、数据存储等方面的差异性，网络孤岛普遍存在，如何解决异构网络所带来的数据共享问题一度成为研究者们面临的极大挑战。不过随后美国国家技术标准局（NIST）和IEEE共同组织了关于制订智能传感器接口和连接网络通用标准的研讨会，产生了IEEE1451传感器/执行器、智能变送器接口标准协议族，试图解决传感器市场上总线不兼容的问题。2005年，开放地理空间联盟（OGC）提出了一种新型的传感器Web整合框架标准，让用户能透过Web的界面来进行节点搜寻、数据获取及节点控制功能。

文献[12]对无线传感器网路的路由协议进行了研究，指出多路径路由发展的趋势和挑战，而文献[13]则从生物学、商业、环境、医疗、工业以及军事等领域探讨无线传感器的重要用途。

（2）Web2.0技术

“Web 2.0”的概念2004年始于出版社经营者O'Reilly和MediaLive International之间的一场头脑风暴论坛，所谓的Web2.0是指互联网上的每一个用户的身份由单纯的“读者”进化为了“作者”以及“共同建设人员”，由被动地接收互联网信息向主动创造互联网信息发展。Web2.0伴随着博客、百科全书以及社交网络等多种应用技术的发展，大量的网页点击与交流促使了大数据的形成，给人类日常生活方式带来了极大的变革。

（3）条形码技术

条形码的使用给零售业带来了革命性的改变，通过内嵌ID等信息，条形码在被扫描之后，快速在数据库中进行ID匹配，便很快就获知该产品的价格、性能、产商等具体信息，条形码被广泛应用于零售商店的收银以及车站售票等业务中，每天大量的商品销售记录通过扫描条形码而产生。近年来的智能手机的盛行，手机应用如微信中的二维条形码也随处可见，文献[14]中设计了一种应用于手机应用的彩色二维条形码，改善了用户对应用程序的感受。

（4）RFID技术

RFID与条形码相比，扩展了操作距离，且标签的使用比条形码容易，携带一个可移动的阅读器便可收集到标签的信息，被广泛应用于仓库管理和清单控制方面。RFID标签可以分为两类，一类是被动的，如今被广泛使用，其造价便宜，但是没有内部电源，依靠阅读器的射频波产生能量，操作距离也很近，因而其适用性也受到了制约；另一类是主动的，其拥有内部电源，因此造价较贵，但是操作距离远，存储能力强，因而适用范围广，在未来这种标签会受到普遍欢迎的。

学术界在RFID技术的研究上已经取得巨大的进步。较早的工作重心大多集中在对标签进行搜集的问题上，即尽可能快地在大量标签中搜集他们的ID，而这方面最大的挑战是解决多标签同时竞争较窄的信道引起冲突的问题。研究者们提出了两类解决思路，即基于ALOHA的协议[15-17]和基于树的协议[18-20]。而其他的工作专注于标签评估问题，即使用统计学的方法来评估一个庞大系统中的标签数目[21-23]。总之，RFID由于具有操作范围广泛、性能稳定以及高存储能力等特性，在工业界中将具有巨大的潜力。

（5）移动终端技术

随着科学技术的发展，移动终端诸如手机、笔记本、平板电脑等随处可见，加上网络的宽带化发展以及集成电路的升级，人类已经步入了真正的移动信息时代。

如今的移动终端已经拥有极强的处理能力，通信、定位以及扫描功能应有尽有，大量的移动软件程序被

开发并应用，人们无时无刻不在接收和发送信息。

目前，智能手机等移动设备的数量仍然在迅猛增长中，移动社交网络也会日益庞大和复杂，海量的数据穿梭其中，针对移动数据的处理也将越来越复杂。

2.3 文件系统

文件系统是支撑上层应用的基础，本小节将简要介绍面向大数据处理的文件系统如谷歌分布式文件系统（GFS），以及一些其他的分布式文件系统。

2.3.1 分布式文件系统GFS

谷歌开发的文件系统GFS[24]，是一个基于分布式集群的大型的分布式文件系统，它为MapReduce计算框架提供底层数据存储和数据可靠性。GFS采用廉价普通磁盘，并把磁盘数据出错视为常态，其自动多数据备份存储也增加了可靠性。

GFS基本构架中，GFS Master保存了GFS文件系统的3种元数据：命名空间、Chunk与文件名的映射表、Chunk副本的位置信息，前两个数据通过操作日志提供容错处理能力，第3个数据存储在Chunk Server上，可在Master失效时快速恢复Master上的元数据；GFS ChunkServer是用来保存大量实际数据的数据服务器。

GFS基本工作过程如下：

（1）在程序运行前，数据已经存储在GFS文件系统中，程序执行时应用程序会告诉GFS Server所要访问的文件名或者数据块索引是什么。

（2）GFS Server根据文件名和数据块索引在其文件目录空间中查找和定位该文件或数据块，并将这些位置信息回送给应用程序。

（3）应用程序根据GFSServer返回的具体Chunk数据块位置信息，直接访问相应的Chunk Server。

（4）应用程序直接读取指定位置的数据进行计算处理。

后来谷歌对GFS进行了改进，并对新版本命名为Colosuss，主要对原有的单点故障、海量小文件存储等诸多问题进行了修正和改进，使得系统更加安全和健壮。

2.3.2 其他文件系统

除了谷歌的GFS，业界其他针对大数据存储需求的文件系统也层出不穷。

Hadoop的文件系统HDFS[25]作为模仿GFS的开源实现，同样也为Hadoop的底层数据存储支撑，提供数据的高可靠性和容错能力，拥有良好的扩展性和高速数据访问性。

SUN公司开发的Lustre[26]是一个大规模的、安全可靠的、具备高可用性的开源集群文件系统，美国能源部在此基础上实现了新一代的集群系统，显著提高了输入输出速度，已在高校、国家实验室和超级计算研究中心产生了深远影响。

Facebook推出的针对海量小文件的文件系统Haystack[27]有效地解决了海量图片存储问题，它实现多个逻辑文件共享一个物理文件功能，并且增加缓存层，部分元数据直接被加载到了内存。

2.4 数据库系统

2.4.1 并行数据库

并行数据库起源于20世纪80年代，并且在不断发展和创新，高性能和高可用性是其最终的目标和优势。

并行数据库通过简单易用的结构化查询语言（SQL）向外提供数据访问服务，加上在索引、数据压缩、可视化等技术方面的不断扩展，使其具有了高性能的优势。但是诸多因素导致了其扩展性面临严峻的挑战，主要体现在：

（1）单机方面，并行数据库基于高端硬件设计，认为查询失败是特例且纠错复杂，不符合大规模集群失效常态的特性。

（2）集群方面，并行数据库对异构网络支持有限，各节点性能不均，容易引起“木桶效应”。

总之，并行数据库的扩展性方面的缺陷使其面临大数据的处理往往力不从心。

2.4.2 MapReduce分布式数据库BigTable

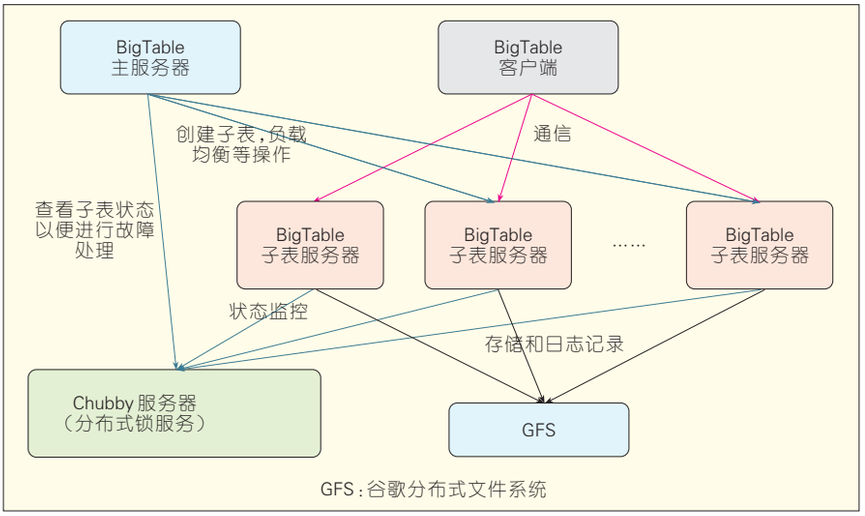
由前述知，并行数据库由于扩展性方面的缺陷无法胜任大数据的处理工作，以谷歌公司推出的BigTable为代表的未采用关系模型的NoSQL（Not only SQL）数据库由此诞生，NoSQL数据库具有模式自由、备份简易、接口简单和支持海量数据等特性，对于实现大数据的存储和处理十分有效。

谷歌在其文件系统之上又设计了MapReduce的分布式数据库BigTable[28]，为应用程序提供了比单纯地文件系统更方便、更高层的数据操作能力，BigTable提供了一定粒度的结构化数据操作能力，主要解决一些大型媒体数据（Web文档、图片等）的结构化存储问题。

BigTable主要是一个分布式多维表，表中数据通过行关键字、列关键字和时间戳来进行索引和查询定位，并且BigTable对存储在表中的数据不做任何解释，一律视为字串，具体数据结构的实现由用户自行定义。

BigTable的基本构架如图3所示，BigTable中的数据均以子表形式保存在子表服务器上，最终以GFS文件形式存储在文件系统中。客户端程序直接和子表服务器通信，Chubby服务器完成对子表服务器的状态监控，主服务器通过查看Chubby服务器目录来终止出现故障的子服务器并将其数据转移至其他子服务器。另外，主服务器还完成子表的创建和负载均衡等操作。

当然，由于MapReduce将本来应由数据库管理系统完成的诸如文件存储格式的设计、模式信息的记录、



▲图 3 BigTable 基本构架图

数据处理算法的实现等工作转移给了程序员，从而导致程序员负担过重。另外，MapReduce 是面向非结构化的大规模数据处理的，往往是一次处理，因而同等硬件条件下的性能也比并行数据库低[29]。

2.4.3 数据库的深层探讨

并行数据库具有高性能的优势，但扩展性问题阻碍了其在大数据处理上的进一步发展，而 MapReduce 性能和易用性上提升空间较大，因此目前两种方案均不理想。业界经过长时间的探讨，基本一致认为并行数据库和 MapReduce 各取其长，相互融合，也许是一种不错的道路[30]。由此诞生了并行数据库主导型、MapReduce 主导型以及并行数据库与 MapReduce 集成型 3 类大数据处理数据库。

（1）并行数据库主导型

这类数据库的基本思路是在并行数据库上增加 MapReduce 的大数据处理能力，将数据分析过程转移到数据库内进行，使得原系统同时获得 SQL 的易用性与 MapReduce 的开放性。但是，并行数据库的扩展能力与容错能力并未得到改善，典型的系统如 Greenplum[31]、Asterdata[32]等。

（2）MapReduce 主导型

这类数据库的基本思路是利用关系数据库的 SQL 接口和模式支持技术改善 MapReduce 的易用性。通过 SQL 接口，可以很简便的完成查询分析等操作，大大减轻了程序员的负担，但 MapReduce 的性能方面仍有待提升，比较典型的系统如 Facebook 的 Hive[33]和 Yahoo！的 Pig Latin[34]等。

（3）并行数据库与 MapReduce 集成型

这类数据库兼顾并行数据库与 MapReduce 的长处，主要分两种思路：按功能将并行数据库与 MapReduce 分别设计到相应的部位以形成一个完整系统，以及整合并行数据库和 MapReduce 这两套完整的系统以构成一个混合系统。

第一种思路典型代表是耶鲁大学提出的 HadoopDB[35]，它将 Hadoop 作为调度层和网络沟通层，关系数据库作为执行引擎，尽可能地将查询压入数据库层处理，Hadoop 框架的应用可以获得较好的容错性和对异构环境的支持，库内数据查询的使用则可获得关系数据库的高性能优势。

第二种思路的代表是 Vertica 数据库[36]，它拥有两套独立完整的系统，Hadoop 负责非结构化数据和耗时的批量复杂数据的处理，Vertica 负责结构化数据的处理以及高性能的交互式查询。

当然，这些思路仍非理想的方案，例如，HadoopDB 丧失了 MapReduce 较低的预处理和维护代价等，Vertica 则依旧存在 Vertica 扩展性问题和 Hadoop 的性能问题。因此，在大数据面前，数据库系统的研究还有很长的路要走，我们在总结传统的数据库经验的同时，还要积极了解新兴的数据库系统，才能更好地促进适应现今大数据发展的性能优良数据库的面世。

2.5 大数据分析技术

用于大数据集的分析方法很多，包括统计学、计算机科学等各个领域的技术。本小节将简要介绍其中几种典型的大数据分析技术，当然，这些技术同样适用于少量数据集的分析，但大数据集环境下的应用无疑会发挥更加明显的作用。

（1）A/B 测试

传统的 A/B 测试，是一种把各组变量随机分配到特定的单变量处理水平，把一个或多个测试组的表现与控制组相比较，进行测试的方式。现在的 A/B 测试主要用于在 Web 分析方面，例如通过对比统计新旧网页的用户转化率，来掌握两种设计的优劣等。大数据时代的到来为大规模的测试提供了便利，提高了 A/B 测试的准确性。由于移动设备及技术的迅猛发展，移动分析也逐渐成为 A/B 测试增长最快的一个领域。

（2）聚类分析

聚类分析指将物理或抽象的集合分组成为由类似的对象组成的多个类的分析过程。聚类分析是一种探索性的数据挖掘分析方法，不需事先给出划分的类的具体情况，主要用在商业、生物学、因特网等多个领域中。对于大数据的分析处理，通过聚类可以简化后续处理过程，并且可以发现其中隐藏的某些规则，充分发挥了大数据的作用。

（3）集成学习

集成学习指的是使用一系列“学习器”进行学习，并使用某种规则把各学习结果进行整合从而获得比单个“学习器”更好的学习效果的一种机器学习方法。对于大数据的集成学习，可以更好地提炼和把握其中的本质属性。

（4）神经网络

神经网络是一种模仿动物神经网络行为特征，进行分布式并行信息处理的算法数学模型，它依靠系统的复杂程度，通过调整内部大量节点之间相互连接的关系，来达到处理信息的目的。

神经网络作为一门新兴的交叉学科，是人类智能研究的重要组成部分，已成为脑科学、神经科学、认知科学、心理学等共同关注的焦点。神经网络对于大数据的并行处理，无疑也是一种比较可行的方式。

（5）自然语言处理

自然语言处理是计算机科学领域与人工智能领域中的一个重要方向，它研究能实现人与计算机之间用自然语言进行有效通信的各种理论和方法。

人与计算机的通信交流往往存在很多歧义，如何消除这些歧义，将带有潜在歧义的自然语言输入转换成某种无歧义的计算机内部表示，是自然语言处理的主要问题。大数据时代意味着有大量的知识和推理来完成消除歧义现象的可能，这也给自然语言处理带来了一些新的挑战和机遇。

大数据分析技术还有很多，例如模式识别、空间分析、遗传算法等等，并且研究者们还在不断地寻找新的更有效地分析方法，另外通过结合多个方法来实现数据分析往往也能达到非常明显的效果。

2.6 大数据的可视化

面对海量的数据，如何将其清晰明朗地展现给用户是大数据处理所面临的巨大挑战。无论是学术界还是工业界，对大数据进行可视化的研究从未停止。通过将大数据图形化、图像化以及动画化等展示出来的技术和方法不断出现，本节将介绍几种典型的案例。

（1）宇宙星球图

俄罗斯工程师 Ruslan Enikeev 根据 2011 年底的互联网数据，将 196 个国家的 35 万个网站数据整合起来，并根据 200 多万个网站链接将这些“星球”通过“关系链”联系起来组成了因特网的“宇宙星球图”[37]。不同颜色代表不同的国家，每个“星球”的大小根据其网站流量来决定，而“星球”距离远近根据链接出现的频率、强度等决定。类似地，对于具有复杂结构的社交网络，“宇宙星球图”同样也十分适用，可以根据个人的知名度、人与人之间的联系等进行绘画星球图。

（2）标签云

“标签云”的设计思路主要是，对于不同的对象用标签来表示，标签的排列顺序一般依照字典排序，按照热门程度确定字体的大小和颜色。例如对于某个文档，出现频度越高的单词将会越大，反之越小。这样，便可以根据字母表顺序和字体的大小来对各单词的具体情况一目了然。文献[38]通过将地图上的各个物理位置根据描述的具体程度用“标签云”表示，使得用户对各个场所的知名程度有个清晰的认识。

（3）历史流图

文献[39]提出了一种用于可视化文档编辑历史的“历史流图”，对于一个面向大众的开放文档，编辑和查阅都是自由的，用户可以随时自由的对文档进行增加或删除操作。“历史流图”中，横坐标轴表示时间，纵坐标轴表示作者，不同作者的不同内容对应中间部分不同颜色和长度，随着时间的推移，文档的内容不断变化，作者也在不断增加中。通过对“历史流图”的观察，很容易看出各人对该文档的贡献，当然，除了发现有人对文档给出有益的编辑外，也存在着一些破坏文档、删除内容的人，但总有逐渐被修复回去的规律。像维基百科等的词条注释文档，“历史流图”的可视化效果十分明显。

关于大数据可视化的方面努力还有很多，不同的“源数据”有不同的可视化策略，大数据可视化的研究工作仍有待进行下去。

3 大数据应用所面临的问题

大数据时代面临的首要问题是人力和财力问题，IDC 分析称，大数据相关人才的欠缺将会成为影响大数据市场发展的一个重要因素。据调查，仅美国就缺少大约 14 万到 19 万的具有深层次数据分析技巧的专业技术人员以及 150 万针对大数据的经理人。据阿里巴巴称，虽然其各类业务产生的数据为数据分析创造了非常好的基础条件，然而却招聘不到合适的数据科学家而影响了研发进展。

高德纳公司预测，到 2015 年，全球将新增 440 万个与大数据相关的工作岗位，且会有 25% 的组织设立首席数据官职位。其中有 190 万个工作岗位将在美国，每一个与大数据有关的 IT 工作，都将在技术行业外部再建 3 个工作岗位，这将在美国再创建将近 600 万个工作岗位。数据科学家是复合型人才，是对数学、统计学、机器学习等多方面知识的综合掌控，能对数据做出预测性的、有价值的分析。因此，各国对大数据人才的培养工作应当快速有效地着手执行。大数据的接收和管理也需要大量的基础设施和能源，无论是传感器还是数据中心的服务器，都需要大量的硬件投入和能源消耗，这也就意味着大数据处理的财力需求极为可观。如何处理好大数据产生的资金投入比例，也成为了各国和各企业决策者面临的难题。

另外，大数据还将面临严重的安全和隐私问题。首先，随处可见的传

感器和摄像头等设备，会监视并记录人们位置等信息，通过海量数据的分析，便可轻易了解人们的行踪规律，从而可能给人们带来生命和财产安全；其次，“云设施”的经济划算，推动了僵尸网络的发展及海量并行处理破解密码系统的可能性；最后，由于云计算要求我们放弃自主计算能力，当整个社会的信息，包括个人信息、商业信息都存储在巨头们提供的“云”上时，我们只能寄希望于这些巨头们都是道德高尚的圣人，否则我们将面临灾难性损失。面对这些安全威胁，学术界和工业界也都纷纷提出自己策略。

针对基于位置服务的安全性问题，文献[40]提出了一种 k–匿名方法，即将自己与周围 k–1 个用户组成一个范围集合性对象来请求位置服务，从而模糊了自己的准确位置。文献[41]提出的策略是，搜集周围的 k–1 个用户的位置信息，并以其中的某一个的名义发送位置服务请求，从而也达到隐藏准确坐标的目的。Roy等人将集中信息流控制和差分隐私保护等技术融入云中的数据生成与计算阶段，提出了一种隐私保护系统Airavat[42]，防止MapReduce计算过程中将非授权的隐私数据泄露出去，并且支持对计算结果的自动除密。Mowbray等人在数据存储和使用阶段使用一种基于客户端的隐私管理工具[43]，提供以用户为中心的信任模型，帮助用户控制自己的敏感信息在云端的存储和使用。

苹果最近申请了一项专利，叫做电子分析污染技术，能够将用户在苹果产品上产生的行为数据进行污染和混淆，让其他厂商获取不到真正的用户数据。这类信息安全保护的思路是：当各种加密措施无法彻底保护个人信息时，不如将大量的垃圾信息、错误信息充斥在真实有效的信息之中，让窃取者不得不耗费巨大的成本从中分析。高德纳公司分析指出，大数据安全是一场必要的斗争，并且大数据本身更可用来提高企业安全。因为解决安全问题的前提是，企业必须先确定正常、非恶意活动是啥样子的，然后查找与之不同的活动；从而，发现恶意活动，基于大数据来建立一个基线标准就很好地达到了这个目的。

最后，大数据的出现会促使IT相关行业的生态环境和产业链的变革。传统的网络公司运营模式是在自己的服务器上来管理若干产品和服务，并通过网络连线提供给用户终端，产生的数据归公司独有。然而，在大数据时代，这种模式已经难以胜任，服务公司往往会选择租赁第三方的开放平台来运营自己的业务。这样，用户提供数据，服务方处理数据，但数据的实际存储地却在第三方。大数据影响的IT产业链大致包括数据资源、应用软件、基础设施三大部分。数据资源方面，各大信息中心、通信运营商等积极研制和引用大数据技术，挖掘大量数据分析相关人才，数据资源的收集和开发产业逐步完善；应用软件方面，随着高性能云平台的出现，云应用软件也不断被开发出来，用户再也不必烦恼复杂的软件安装和配置过程，便可以轻松享受各种网络应用服务；基础设施方面，大数据对硬件的依赖，迫使高性能硬盘、低能耗服务器、小巧化个人终端等行业的快速发展。另外，大数据技术的日益成熟也会促使跨行业经营模式的发展。第三方可以将用户的各种服务请求进行打包，然后利用大数据分析来寻求最好的服务商的组合以反馈给用户。对服务提供方来说，借助第三方可以更好地推销自己的服务。而对第三方而言，可以获得大量的分析数据，其中的利益也是可观的，真正的实现了“双赢”，同时也使得用户获得更好的服务体验。

4 结束语

大数据时代挑战与机遇并存，正确处理好大数据，不仅符合企业的利益，也给人们日常生活带来极大的便利。本文对大数据的基本概念、处理流程以及相关技术进行了简要的探讨，并分析了大数据可能带来的一些问题及应对策略。云计算目前是处理大数据的基础技术，但其在安全和隐私方面的保障工作仍让不少人感到怀疑，根本原因还是个人和商业的信息都存放在远端的巨头们提供的看不见的“云”上。大数据时代已经到来，但是，相应的技术体系和社会保障仍是亟需研究的应用课题。

参考文献

[1] MANYIKA J. Big data: The next frontier for innovation, competition, and productivity [R]. Executive Summary, McKinsey Global Institute, 2011.
[2] LI T, CHEN S, LING Y. Identifying the missing tags in a large RFID system [C]//Proceedings of the 11th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc'10),Sept 20–24, 2010, Chicago, IL, USA. New York, NY, USA: ACM, 2010:10p.
[3] BOHLOULI M, SCHULZ F, ANGELIS L, et al. Towards an integrated platform for big data analysis [C]//Proceedings of the International Conference of Integrated Systems Design and Technology (ISDT'12), May 16–18,2012, Mallorca, Spain. Berlin, Germany: Springer–Verlag, 2013:47–56.
[4] IBM. bringing big data to the enterprise [EB/OL]. [2013–02–05]. http://www–01.ibm.com/software/data/bigdata/.
[5] Nature. BigData [EB/OL]. [2012–10–02]. http://www.nature.com/news/specials/bigdata/index.html.
[6] Big Data Across the Federal Government [EB/OL]. [2012–10–02], http://www.whitehouse.gov/sites/default/files/microsites/ostp/big_data_fact_sheet_final_1.pdf.
[7] DEAN J, GHEMAWAT S. MapReduce: Simplified data processing on large clusters [C]//Proceedings of the 6th USENIX Symposium on Operating System Design and Implementation (OSDI'04), Dec 6–8,2004, San Francisco, CA,USA. Berkeley, CA, USA: USENIX Association, 2004:137–150.
[8] GENOVESE Y, PRENTICE S. Pattern–based strategy: Getting value from big data [R]. Gartner Inc, 2011.
[9] LABRINIDIS A, JAGADISH H V. Challenges and opportunities with big data [J]. Proceedings of the VLDB Endowment(PVLDB), 2012,5(12):2032–2033.
[10] WANG L, TAO J, KUNZE M. Scientific cloud computing: early definition and experience [C]//Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications(HPCC'08), Sept 25–27, 2008, Dalian, China. Piscataway, NJ,USA: IEEE, 2008:825–830.
[11] FOSTER I, ZHAO Y, RAICU I, et al. Cloud computing and grid computing 360–degree

compared [C]//Proceedings of the Grid Computing Environments Workshop(GCE'08), Nov 12–16,2008, Austin, TX,USA. Piscataway, NJ,USA: IEEE, 2008:10p.
[12] RADI M, DEZFOULI B, BAKAR K A. Multipath routing in wireless sensor networks: Survey and research challenges [J]. Sensors, 2012, 12(1):650–685.
[13] GILBERT E P K, KALIAPERUMA B L. Research issues in wireless sensor network applications: A survey [J]. International Journal of Information and Electronics Engineering, 2012,2(5):702–706.
[14] ZHAI J, WANG G N. An anti–collision algorithm using two–functioned estimation for RFID tags [C]//Proceedings of the International Conference on Computational Science and Its Applications (ICCSA'05):Vol 4, May 9–12,2005, Singapore. LNCS 3480. Berlin, Germany: Springer–Verlag, 2005: 702–711.
[15] CHA J, KIM J. Novel anti–collision algorithms for fast object identification in RFID system [C]//Proceedings of the 11th International Conference on Parallel and Distributed Systems (ICPADS'05):Vol 2,Jul 20–22,2005, Fuduoka, Japan . Los Alamitos, CA,USA: IEEE Computer Society, 2005: 63–67.
[16] VOGT H. Efficient object identification with passive RFID tags [C]//Proceedings of the 1st International Conference on Pervasive Computing(Pervasive'02), Aug 26–28, 2002, Zurich, Switzerland. Berlin, Germany: Springer–Verlag, 2002:98–113.
[17] HUSH D, WOOD C. Analysis of tree algorithm for RFID arbitration [C]// Proceedings of the 1998 IEEE International Symposium on Information Theory(ISIT'98), Aug 16–21,1998, Cambridge, MA, USA . Piscataway, NJ,USA: IEEE, 1998.
[18] MYUNG J, LEE W. An adaptive memoryless tag anti–collision protocol for RFID networks [C]//Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'05):Vol 3, Mar 13–17, 2005, Miami, FL, USA. Piscataway, NJ,USA: IEEE, 2005.
[19] CHOI H, CHA J, KIM J. Fast wireless anti–collision algorithm in ubiquitous ID system [C]//Proceedings of the 60th Vehicular Technology Conference (VTC–Fall'04): Vol 6, Sep 26–29,2004, Los Angeles, CA,USA. Piscataway, NJ, USA: IEEE, 2004: 4589–4592 .
[20] KODIALAM M, NANDAGOPAL T. Fast and reliable estimation schemes in RFID systems [C]//Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MOBICOM' 06), Sept 24–29,2006, Los Alamitos, CA, USA. New York, NY,USA: ACM, 2006: 322–333.
[21] KODIALAM M, NANDAGOPAL T, LAU W. Anonymous tracking using RFID tags [C]// Proceedings of the 26th Annual Joint Conference of the IEEE Computer and Communications (INFOCOM'07), May 6–12,2007, Anchorage, AK,USA. Piscataway, NJ,USA: IEEE, 2007: 1217–1225.
[22] QIAN C, NGAN H, LIU Y. Cardinality estimation for large–scale RFID systems [C]//Proceedings of the 6th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom'08), Mar17–21, 2008, Hong Kong, China. Piscataway, NJ,USA: IEEE, 2008:30–39.
[23] GHEMAWAT S, GOBIOFF H, LEUNG S T. The Google file system [C]//Proceedings of the 19th ACM SIGOPS Symposium on Operating Systems Principles (SOSP'03), Oct 19–22, 2003, Bolton Landing, NY, USA. New York, NY, USA: ACM, 2003:29–43.
[24] HDFS Architecture Guide [EB/OL]. [2013–01–08]. http://archive.cloudera.com/cdh4/cdh/4/mr1/hdfs_design.pdf.
[25] Lustre [EB/OL]. [2013–02–12]. http://www.lustre.org.
[26] BEAVER D, KUMAR S, LI H C, et al. Finding a needle in haystack: Facebook's photo storage [C]//Proceedings of the 9th USENIX Symposium on Operating System Design and Implementation (OSDI'10),Oct 4–6, 2010, Vancouver, Canada. Berkeley, CA, USA: USENIX Association,2010:47–60.
[27] CHANG F, DEAN J, GHEMAWAT S, et al. Bigtable: A distributed storage system for structured data. [C]//Proceedings of the 7th USENIX Symposium on Operation Systems Design and Implementation (OSDI'06), Nov 6–8,2006, Seattle, WA, USA. Berkeley, CA, USA: USENIX Association, 2006:205–218.
[28] PAVLO A, RASIN A, MADDEN S, et al. A comparison of Approaches to large scale data analysis [C]//Proceedings of the 35th ACM SIGMOD International Conference on Management of Data(SIGMOD'09), Jun 29–Jul 2, 2009,Providence, Rhode Island. New York, NY, USA:ACM, 2009:165–178.
[29] STONEBRAKER M, ABADI D , DEWITT D J, et al. MapReduce and parallel DBMSs: Friends or foes? [J]. Communications of the ACM, 2010,53(1):64–71.
[30] Greenplum MapReduce [EB/OL]. [2012–12–21]. http://www.greenplum.com/technology/MapReduce.
[31] Asterdata MapReduce [EB/OL]. [2012–12–21]. http://www.asterdata.com/resources/MapReduce.php.
[32] Hive[EB/OL]. [2012–12–21]. http://hive.apache.org/.
[33] OLSTON C, REED B, SRIVASTAVA U, et al. Pig Latin: A not–so–foreign language for data processing [C]//Proceedings of the 34th ACM SIGMOD International Conference on Management of Data(SIGMOD'08), Jun 9–12, 2008, Vancouver, Canada. New York, NY, USA: ACM, 2008: 1099–1110.
[34] ABOUZEID A, BAJDA–PAWLIKOWSKI K, ABADI D J, et al. HadoopDB: An architectural hybrid of MapReduce and DBMS technologies for analytical workloads [C]//Proceedings of the 35th International Conference on Very Large Data Bases (VLDB'09), Aug 24–28, 2009, Lyon, France. New York, NY,USA:ACM, 2009: 922–933.
[35] Vertica [EB/OL]. [2012–11–03]. http://www.vertica.com/the–analytics–platform/native–bi–etl–and–hadoop–MapReduce–integration/.
[36] The Internet Map [EB/OL]. [2012–12–18]. http://internet–Map.net/.
[37] PAELKE V, DAHINDEN T, EGGERT D, et al. Location based context awareness through tag–cloud visualizations [C]//Proceedings of the Joint International Conference on Theory, Data Handling and Modelling in GeoSpatial Information Science(ISGIS'10), May 26–28,2010, Hong Kong, China. New York, NY, USA:ACM, 2010 290–295.
[38] VIÉGAS F B, WATTENBERG M, DAVE K. Studying cooperation and conflict between authors with history flowvisualizations [C]// Proceedings of the ACM Conference on Human Factors in Computing Systems(CHI'04), Apr 24–29, 2004, Vienna, Austria. New York, NY, USA:ACM, 2004:575–582.
[39] SWEENEY L. k–anonimity: A model for protecting privacy [J]. International Journal of Uncertainty, Fuzziness and Knowledge–based Systems, 2002, 10(5): 557–570.
[40] DOMINGO–FERRER J. Micro aggregation for database and location privacy [C]//Next Generation Information Technologies and Systems: Proceedings of the 6th International Workshop on Next Generation Information Technologies and Systems(NGITS'06), Jul 4–6,2006,Kibbutz Shefayim, Israel. LNCS 4032. Berlin, Germany: Springer–Verlag, 2006:106–116.
[41] ROY I, RAMADAN H E, SETTY S T V, et al. Airavat: Security and privacy for MapReduce [C]//Proceedings of the 9th USENIX Symposium on Operation Systems Design and Implementation (OSDI'10), Oct 4–6, 2010, Vancouver, Canada. Berkeley, CA, USA: USENIX Association, 2010.297–312.
[42] BOWERS K D, JUELS A, OPREA A. Proofs of retrievability: Theory and implementation [C]//Proceedings of the 1st ACM Workshop on Cloud Computing Security Workshop(CCSW'09), Nov 13,2009, Chicago, IL,USA. New York, NY,USA: ACM, 2009:43–54.
[43] CHEN Z J, ZHAO Y, LIN C, et al. Accelerating large–scale data distribution in booming Internet: Effectiveness, bottlenecks and practices [J]. IEEE Transactions on Consumer Electronics, 2009,55(2):518–526.

作者简介

窦万春，南京大学计算机科学与技术系、南京大学软件新技术国家重点实验室教授、博士生导师；主要从事云计算、服务计算等方面的研究工作；已主持或参与完成基金项目8项已发表学术论文60余篇。

江澄，南京大学计算机科学与技术系在读硕士研究生；研究方向为服务计算、云计算等。

DOI: 10.3969/j.issn.1009-6868.2013.04.003
网络出版地址：http://www.cnki.net/kcms/detail/34.1228.TN.20130627.0859.001.html

大数据关键技术
Key Big-Data Technologies

中图分类号：TN915.03; TP393.03　文献标志码：A　文章编号：1009-6868 (2013) 04-0017-005

王秀磊/WANG Xiulei
刘鹏/LIU Peng
（解放军理工大学 指挥信息系统学院，江苏 南京 210007）
(College of Command Information Systems, PLA University of Science & Technology, Nanjing 210007, China)

摘要： 结合大数据系统的一般结构，介绍和对比了当前大数据领域在文件存储、数据处理和数据库领域的关键技术。通过各种技术的对比，得到了一些分析结果。分析结果表明大数据系统的解决方案必将落地于现有的云计算平台；云计算平台的分布式文件系统、分布式运算模式和分布式数据库管理技术是解决大数据问题的基础；一些大的依靠数据盈利的大公司必然会是大数据应用的主体。

关键词： 大数据；分布式文件系统；分布式数据库；MapReduce 技术

Abstract: In this paper, we discuss the general structure of a big-data system as well as key technologies in big-data storage, processing, and database. We compare these technologies in order find problems in the big-data system and propose solutions that will be used in the cloud computing platform. We propose distributed file system, computing model, and database management to solve problems associated with big data. Big companies that profit from big data will be the main users of big-data applications.

Key words: big data; distributed file system; distributed database; MapReduce

21世纪，世界已经进入数据大爆炸的时代，大数据时代已经来临。从商业公司内部的各种管理和运营数据，到个人移动终端与消费电子产品的社会化数据，再到互联网产生的海量信息数据等，每天世界上产生的信息量正在飞速增长。2009年数据信息量达到8 000亿GB，而到2011年达到1.8 ZB[1]。图灵奖获得者Jim Gray提出的“新摩尔定律”：“每18个月全球新增信息量是计算机有史以来全部信息量的总和”，已经得到验证。

大数据的“大”不仅仅体现在数据的海量性，还在于其数据类型的复杂性。随着报表、账单、影像、办公文档等在商业公司中得到普遍使用，互联网上视频、音乐、网络游戏不断发展，越来越多的非结构化数据进一步推动数字宇宙爆炸。数据海量而复杂，这是对大数据的诠释。与传统的数据相比，大数据具有规模性（Volume）、多样性（Variety）、高速性（Velocity）和低价值密度（Value）的4V特点[2]。规模性和高速性是数据处理一直以来研究和探讨的问题，多样性和价值密度低是当前数据处理发展中不断显现出来的问题，而且在可以预见的未来，随着智慧城市、智慧地球等各种新设想的不断成为现实，上面的4中问题将会变得更加凸显，而且是不得不面对的问题。

数据的产生经历了被动、主动和自动3个阶段[3]。大数据的迅猛发展是信息时代数字设备计算能力和部署数量指数增长的必然结果。解决大数据研究中的问题，必须要从大数据的产生背景进行研究。大数据的产生源于规模效应，这种规模效应给数据的存储、管理以及数据的分析带来了极大的挑战，数据管理方式上的变革正在酝酿和发生。大数据的规模效应要求其存储、运算方案也应当从规模效应上进行考虑。传统的单纯依靠单设备处理能力纵向发展的技术早已经不能满足大数据存储和处理需求。以Google等为代表的一些大的数据处理公司通过横向的分布式文件存储、分布式数据处理和分布式的数据分析技术很好的解决了由于数据爆炸所产生的各种问题。

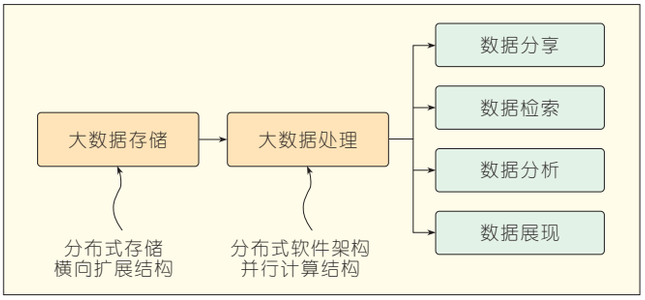
1 大数据关键技术

1.1 大数据系统的架构

大数据处理系统不管结构如何复杂，采用的技术千差万别，但是总体上总可以分为以下的几个重要部分。大数据系统结构如图1所示。

从数据处理的一般流程可以看到，在大数据环境下需要的关键技术主要针对海量数据的存储和海量数据的运算。传统的关系数据库经过近40年的发展已经成为了一门成熟

收稿日期：2013-04-15
网络出版时间：2013-06-27
基金项目：国家科技重大专项（2012ZX03002003）



◀图1
大数据系统结构

同时仍在不断演进的数据管理和分析技术，结构化查询语言（SQL）作为存取关系数据库的语言得到了标准化，其功能和表达能力也得到的不断增强。但是，关系数据管理系统的扩展性在互联网环境下遇到了前所未有的障碍，不能胜任大数据分析的要求。关系数据管理模型追求的是高度的一致性和正确性。纵向扩展系统，通过增加或者更换CPU、内存、硬盘以扩展单个节点的能力，终会遇到“瓶颈”。

大数据的研究主要来源于依靠数据获取商业利益的大公司。Google公司作为全球最大的信息检索公司，其走在了大数据研究的前沿。面对呈现爆炸式增加的因特网信息，仅仅依靠提高服务器性能已经远远不能满足业务的需求。如果将各种大数据应用比作“汽车”，支撑起这些“汽车”运行的“高速公路”就是云计算。正是云计算技术在数据存储、管理与分析等方面的支持，才使得大数据有用武之地。Google公司从横向进行扩展，通过采用廉价的计算机节点集群，改写软件，使之能够在集群上并行执行，解决海量数据的存储和检索功能。2006年Google首先提出云计算的概念。支撑Google公司各种大数据应用的关键正是其自行研发的一系列云计算技术和工具。Google公司大数据处理的三大关键技术为：Google文件系统GFS[4]、MapReduce[5]和Bigtable[6]。Google的技术方案为其他的公司提供了一个很好的参考方案，各大公司纷纷提出了自己的大数据处理平台，采用的技术也都大同小异。下面将从支持大数据系统所需要的分布式文件系统、分布式数据处理技术、分布式数据库系统和开源的大数据系统Hadoop等方面介绍大数据系统的关键技术。

1.2 分布式文件系统

文件系统是支持大数据应用的基础。Google是有史以来唯一需要处理如此海量数据的大公司。对于Google而言，现有的方案已经难以满足其如此大的数据量的存储，为此Google提出了一种分布式的文件管理系统——GFS。

GFS与传统的分布式文件系统有很多相同的目标，比如，性能、可伸缩性、可靠性以及可用性。但是，GFS的成功之处在于其与传统文件系统的不同。GFS的设计思路主要基于以下的假设：对于系统而言，组件失败是一种常态而不是异常。GFS是构建于大量廉价的服务器之上的可扩展的分布式文件系统，采用主从结构。通过数据分块、追加更新等方式实现了海量数据的高效存储，如图2所示给出了GFS体系结构。但是随着业务量的进一步变化，GFS逐渐无法适应需求。Google对GFS进行了设计，实现了Colosuss系统，该系统能够很好地解决GFS单点故障和海量小文件存储的问题。

除了Google的GFS，众多的企业和学者也从不同的方面对满足大数据存储需求的文件系统进行了详细的研究。微软开发的Cosmos[7]支撑其搜索、广告业务。HDFS[8]、FastDFS[9]、OpenAFS[10]和CloudStore[11]都是类似GFS的开源实现。类GFS的分布式文件系统主要针对大文件而设计，但是在图片存储等应用场景中，文件系统主要存储海量小文件，Facebook为此推出了专门针对海量小文件的文件系统Haystack[12]，通过多个逻辑文件共享同一个物理文件，增加缓存层、部分元数据加载到内存等方式有效地解决了海量小文件存储的问题。Lustre是一种大规模、安全可靠的，具备高可靠性的集群文件系统，由SUN公司开发和维护。该项目主要的目的就是开发下一代的集群文件系统，可以支持超过10 000个节点，数以拍字节的数量存储系统。

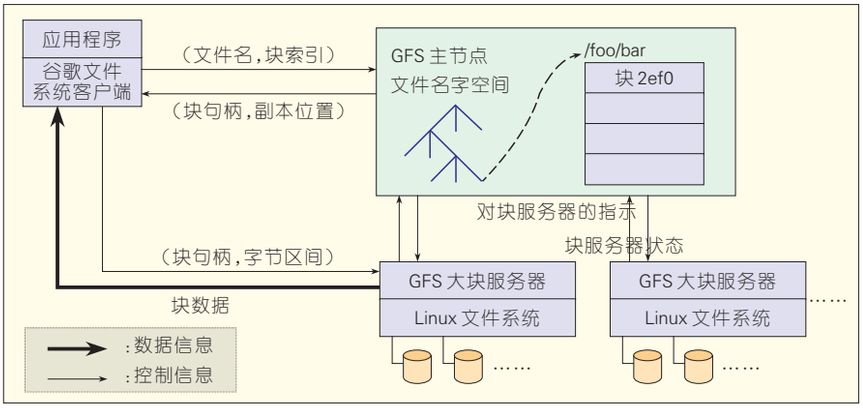
1.3 分布式数据处理系统

大数据的处理模式分为流处理和批处理两种[13-14]。流处理是直接处理，批处理采用先存储再处理。

流处理将数据视为流，源源不断的数据形成数据流。当新的数据到来即立即处理并返回所需的结果。大数据的实时处理是一个极具挑战性的工作，数据具有大规模、持续到达的特点。因此，如果要求实时的处理大数据，必然要求采用分布式的方式，在这种情况下，除了应该考虑分布式系统的一致性问题，还将涉及到分布式系统网络时延的影响，这都增加了大数据流处理的复杂性。目前比较有代表性的开源流处理系统主要有：Twitter的Storm[15]、Yahoo的S4[16]以及Linkedin的Kafka[17]等。

Google公司2004年提出的MapReduce编程模型是最具代表性的批处理模型。MapReduce架构的程序能够在大量的普通配置的计算机上实现并行化处理。这个系统在运行时只关心如何分割输入数据，在大量计算机组成的集群上的调度，集群中计算机的错误处理，管理集群中的计算机之间必要的通信。

对于有些计算，由于输入数据量的巨大，想要在可接受的时间内完成运算，只有将这些计算分布在成百上



▲图 2 GFS 体系结构

千的主机上。这种计算模式对于如何处理并行计算、如何分发数据、如何处理错误需要大规模的代码处理，使得原本简单的运算变得难以处理。MapReduce 就是针对上述问题的一种新的设计模型。

MapReduce 模型的主要贡献就是通过简单的接口来实现自动的并行化和大规模的分布式计算，通过使用 MapReduce 模型接口实现在大量普通的 PC 上的高性能计算。

MapReduce 编程模型的原理：利用一个输入键-值（Key/Value）对集合来产生一个输出的 key/value 对集合。MapReduce 库的用户用两个函数表达这个计算：Map 和 Reduce。用户自定义的 Map 函数接受一个输入的 key/value 值，然后产生一个中间 key/value 对集合。MapReduce 库把所有具有相同中间 key 值的 value 值集合在一起传递给 Reduce 函数。用户自定义的 Reduce 函数接收一个中间 key 的值和相关的一个 value 值的集合。Reduce 函数合并这些 value 值，形成一个较小的 value 值集合，如图 3 所示。

MapReduce 的提出曾经遭到过一系列的指责和诟病。数据专家 Stonebraker 就认为 MapReduce 是一个巨大的倒退，指出其存取没有优化、依靠蛮力进行数据处理等问题。但是随着 MapReduce 在应用上的不断成功，以其为代表的大数据处理技术还是得到了广泛的关注。研究人员也针对 MapReduce 进行了深入的研究，目前针对 MapReduce 性能提升研究主要有以下几个方面：多核硬件与 GPU 上的性能提高；索引技术与连接技术的优化；调度技术优化等。在 MapReduce 的易用性的研究上，研究人员正在研究更为高层的、表达能力更强的语言和系统，包括 Yahoo 的 Pig、Microsoft 的 LINQ、Hive 等。

除了 Google 的 MapReduce，Yunhong Gu 等人设计实现了 Sector and Sphere 云计算平台[18]，包括 Sector 和 Sphere 两部分。Sector 是部署在广域网的分布式系统，Sphere 是建立在 Sector 上的计算服务。Sphere 是以 Sector 为基础构建的计算云，提供大规模数据的分布式处理。Sphere 的基本数据处理模型如图 4 所示。

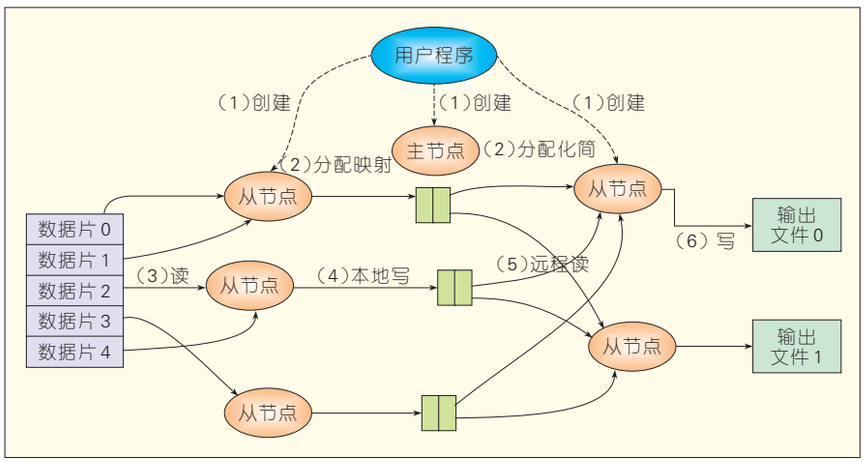
针对不同的应用会有不同的数据，Sphere 统一地将它们以数据流的形式输入。为了便于大规模地并行计算，首先需要对数据进行分割，分割后的数据交给 SPE 执行。SPE 是 Sphere 处理引擎，是 Sphere 的基本运算单元。除了进行数据处理外 SPE 还能起到负载平衡的作用，因为一般情况下数据量远大于 SPE 数量，当前负载较重的 SPE 能继续处理的数据就较少，反之则较多，如此就实现了系统的负载平衡。

1.4 分布式数据库系统

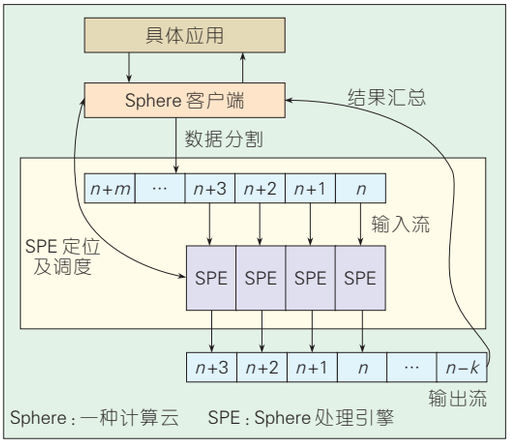
传统的关系模型分布式数据库难以适应大数据时代的要求，主要的原因有以下几点：

（1）规模效应带来的压力。大数据时代的数据远远超出单机处理能力，分布式技术是必然的选择。传统的数据库倾向于采用纵向扩展的方式，这种方式下性能的增加远低于数据的增加速度。大数据采用数据库系统应该是横向发展的，这种方式具有更好的扩展性。

（2）数据类型的多样性和低价值密度性。传统的数据库适合结构清晰，有明确应用目的的数据，数据的价值密度相对较高。在大数据时代数据的存在的形式是多样的，各种半结构化、非结构化的数据是大数据的重要组成部分。如何利用如此多样、海量的低价值密度的数据是大数据



▲图 3 MapReduce 工作流程



▲图 4 Sphere 的基本数据处理模型

时代数据库面临的重要挑战之一。

（3）设计理念的冲突。关系数据库追求的是“一种尺寸适用所有”，但在大数据时代不同的应用领域在数据理性、数据处理方式以及数据处理时间的要求上千差万别。实际处理中，不可能存在一种统一的数据存储方式适应所有场景。

面对这些挑战，Google 公司提出了 Bigtable 的解决方案。Bigtable 的设计目的是可靠的处理拍字节级别的数据，并且能够部署到千台机器上。Bigtable 已经实现了以下几个目标：适用性广泛、可扩展、高性能和高可靠性。Bigtable 已经在超过 60 个 Google 的产品和项目上得到了应用。这些产品在性能要求和集群的配置上都提出了迥异的需求，Bigtable 都能够很好地满足。Bigtable 不支持完整的关系数据模型，为用户提供了简单的数据模型，利用这个模型，客户可以动态控制数据的分布和格式。用户也可以自己推测底层存储数据的位置相关性。数据的下标是行和列的名字，名字可以是任意的字符串。Bigtable 将存储的数据都视字符串，但是 Bigtable 本身不去解释这些字符串，客户程序通常会把各种结构化或者半结构化的数据串行化到这些字符串。通过仔细选择数据的模式，客户可以控制数据的位置的相关性。最后，可以通过 Bigtable 的模式参数来控制数据是存放在内存中、还是硬盘上。Bigtable 数据模型如图 5 所示，给出了 Bigtable 存储大量网页信息的实例。

除了 Google 公司为人熟知的 Bigtable，其他的大型 Internet 内容提供商也纷纷提出大数据系统。具有代表性的系统有 Amazon 的 Dynamo[19] 和 Yahoo 的 PNUTS[20]。Dynamo 综合使用了键/值存储、改进的分布式哈希表（DHT）、向量时钟等技术实现了一个完全的分布式、去中性化的高可用系统。PNUTS 是一个分布式的数据库系统，在设计上使用弱一致性来达到高可用性的目标，主要的服务对象是相对较小的记录，比如在线的大量单个记录或者小范围记录集合的读和写访问，不适合存储大文件、流媒体。

Bigtable、Dynamo、PNUTS 等技术的成功促使研究人员开始对关系数据库进行反思，产生了一批为采用关系模型的数据库，这些方案通称为：NoSQL（not only SQL）。NoSQL 数据库具有以下的特征：模式只有、支持简易备份、简单的应用程序接口、一致性、支持海量数据。目前典型的非关系型数据库主要有以下集中类别，如表 1 所示[21]。

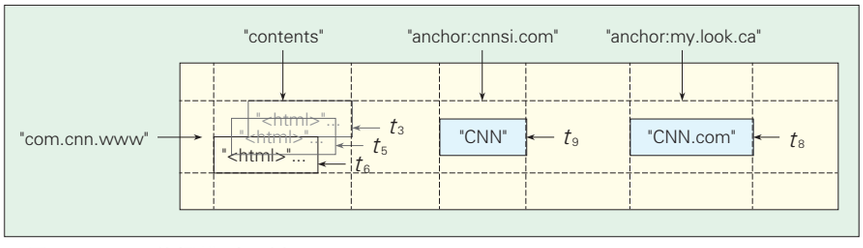
1.5 大数据系统的开源实现平台 Hadoop

除了商业化的大数据处理方案，还有一些开源的项目也在积极的加入到大数据的研究当中。Hadoop[22] 是一个开源分布式计算平台，它是 MapReduce 计算机模型的载体。借助于 Hadoop，软件开发者可以轻松地编出分布式并行程序，从而在计算机集群上完成海量数据的计算。Intel 公司给出了一种 Hadoop 的开源实现方案，如图 6 所示。

在该系统中 HDFS 是与 GFS 类似的分布式文件系统，它可以构建从几台到几千台常规服务器组成的集群，并提供高聚合输入输出的文件读写访问。HBase[23] 是与 Bigtable 类似的分布式、按列存储的、多维表结构的实时分布式数据库。可以提供大数据量结构化和非结构化数据的高度读写操作。Hive[24] 是基于 Hadoop 的大数据分布式数据仓库引擎。它可以将数据存放在分布式文件系统或分布式数据库中，并使用 SQL 语言进行海量信息的统计、查询和分析操作。ZooKeeper[25] 是针对大型分布式系统的可靠协调系统，提供的功能包括：配置维护、名字服务、分布式同步、组服务等。它可以维护系统配置、群组用户和命名等信息。Sqoop[26] 提供高效在 Hadoop 和结构化数据源之间双向传送数据的连接器组件。它将数据传输任务转换为分布式 Map 任务实现，在传输过程中还可以实现数据转换等功能。Flume[27] 是分布式、高可靠的和高可用的日志采集系统，它用来从不同源的系统中采集、汇总和搬移大量日志数据到一个集中式的数据存储中。

2 结束语

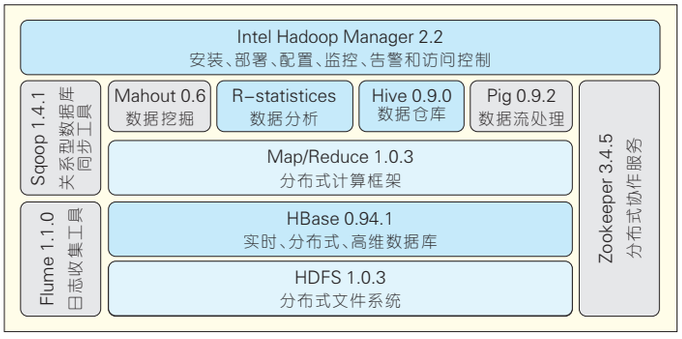
本文结合大数据的产生背景、需求和系统结构，介绍了当前全球在大数据技术方面的进展情况。从分析



▲图 5 Bigtable 数据模型示例

▼表1 典型NoSQL数据库

类别	相关数据库	性能	扩展性	灵活性	复杂性	优点	缺点
Key-Value	Redis Riak	高	高	高	无	查询高效	数据存储缺乏结构
Column	HBase Cassandra	高	高	中	低	查询高效	功能有限
Document	CouchDB MongoDB	高	可变	高	低	对数据结构限制小	查询性能低
Graph	OrientDB	可变	可变	高	高	图算法高效	数据规模小



◀图6 英特尔Hadoop发行版IDH组件

可以看到，大数据系统的解决方案必将落地于现有的云计算平台。云计算平台的分布式文件系统、分布式运算模式和分布式数据库管理技术都为解决大数据问题提供了思路和现成的平台。通过分析也可以看到，大数据的问题的研究，必然是以商业利益为驱动，一些大的依靠数据牟利的大公司必然会是大数据应用的主体，大数据一定会成为的重点领域。总的来说，目前对于大数据的研究仍处于一个非常初步的阶段，还有很多问题需要解决，希望本文的介绍能够给大数据研究的同行提供一定的参考。

参考文献

[1] MANYIKA J, CHUI M, BROWN B, et al. Big data: The next frontier for innovation, competition, and productivity [EB/OL]. [2012–10–02]. http://www.mckinsey.com/Insight/MGI/Research/Technology_and_Innovation/Big_data_The_next_frontier_for_innovation.
[2] BARWICK H. The “four Vs” of big data. Implementing Information Infrastructure Symposium [EB/OL]. [2012–10–02]. http://www.computerworld.com .au/article/396198/iiis_four_vs_big_data/.
[3] 孟小峰, 慈祥. 大数据管理：概念、技术与挑战 [J]. 计算机研究与发展, 2013,50(1):146–169.
[4] GHEMAWAT S, GOBIOFF H, LEUNG S. The Google file system [C]//Proceedings of the 19th ACM SIGOPS Symposium on Operating Systems Principles (SOSP’03), Oct 19 – 22, 2003, Bolton Landing, NY, USA. New York, NY, USA: ACM, 2003:29–43.
[5] DEAN J, GHEMAWAT S. MapReduce: Simplified data processing on large clusters [C]//Proceedings of the 6th USENIX Symposium on Operation Systems Design and Implementation (OSDI’04), Dec 6–8, 2004, San Francisco, CA USA. New York, NY, USA: ACM, 2004:137–150.
[6] CHANG F, DEAN J, GHEMAWAT S, et.al. Bigtable: A distributed storage system for structured data [C]//Proceedings of the 7th USENIX Symposium on Operating Systems Design and Implementation (OSDI’06), Nov 6–8,2006, Seattle,WA, USA. Berkeley, CA, USA: USENIX Association, 2006:205–218.
[7] CHAIKEN R, JENKINS B, LARSON P, et al. SCOPE: Easy and efficient parallel processing of massive data sets [J]. Proceedings of the VLDB Endowment (PVLDB), 2008, 1 (2): 1265–1276.
[8] HDFS Architecture Guide [EB/OL]. [2012–10–02]. http://hadoop.apache.org/docs/hdfs/r0.22.0/hdfs_design.html.
[9] FastDFS [EB/OL]. [2012–10–02]. http://code.google.com/p/fastdfs/w/list.
[10] OpenAFS [EB/OL]. http://www.OpenAFS.org.
[11] CloudStore [EB/OL]. [2012–10–02]. http://code.google.com/p/kosmosfs/.
[12] BEAVER D, KUMAR S, LI H C, et al. Finding a needle in haystack: Facebook’s photo storage [C]//Proceedings of the 9th USENIX Symposium on Operating System Design and Implementation (OSDI’10), Oct 4–6, 2010, Vancouver, Canada. Berkeley, CA, USA: USENIX Association, 2010:47–60.
[13] KUMAR R. Two computational paradigms for big data. KDD summer school [EB/OL]. [2012–10–02]. http://kdd2012. Sigkdd.org/sites/images/summerschool/Ravi–Kumar.pdf.
[14] The big data management challenge [EB/OL]. [2012–10–02]. http://reports.informationweek.com/abstract/81/8766/business–intelligence–and–information–management/research–the–big–data–management–challenge.html.
[15] Storm [EB/OL]. [2012–10–02]. http://github.com/nathanmarz/storm.
[16] NEUMEYER L, ROBBINS B, NAIR A, et al. S4: Distributed stream computing platform. Proceedings of the IEEE International Conference on Data Mining Workshops (ICDMW’10), Dec 14–17,2010, Sydney, Australia. Los Alamitos, CA, USA: IEEE Computer Society, 2010: 170–177.
[17] GOODHOPE K, KOSHY J, KREPS J, et al. Building linkedIn’s real–time activity data pipeline [J]. IEEE Data Engineering Bulletin, 2012,35(2):33–45.
[18] GU Y H, GROSSMAN R. Sector and sphere: The design and implementation of a high performance data cloud [J]. Philosophical Transactions of the Royal Society A, 2009, 367: 2429–2445.
[19] DECANDIA G, HASTORUN D, JAMPANI M, et al. Dynamo: Amazon’s highly available key–value store [C]//Proceedings of the 21th ACM SIGOPS Symposium on Operating Systems Principles (SOSP’07), Oct 14–17, 2007, Washington, DC,USA. New York, NY, USA: ACM, 2007:205–220.
[20] COOPER B F, RAMAKRISHNAN R, SRIVASTAVA U, et al. PNUTS: Yahoo!’s hosted data serving platform [J]. Proceedings of the VLDB Endowment (PVLDB), 2008,1(2):1277–1288.
[21] STRAUCH C. NoSQL databases [EB/OL]. [2012–10–02]. http://www.christof–strauch. De/nosqldbs.pdf.
[22] Hadoop [EB/OL]. [2012–10–02]. http://hadoop.apache.org.
[23] HBase [EB/OL]. [2012–10–02]. http://yankay.com/up–content/hbase/book.html.
[24] Hive [EB/OL]. [2012–10–02]. http://cwiki.apache.org./conflunce/display/Hive/Home.
[25] Zookeeper [EB/OL]. [2012–10–02]. http://zookeeper.apache.org.
[26] Sqoop [EB/OL]. [2012–10–02]. http://spoop.apache.org.
[27] Flume [EB/OL]. [2012–10–02]. http://flume.apache.org.

作者简介

王秀磊，解放军理工大学在读博士研究生；研究方向为容迟/容断网络、软件定义网络、内容中心网络、网络测量和网络管理；已发表学术论文4篇。

刘鹏，清华大学博士毕业；解放军理工大学教授、博导、学科带头人，中国云计算专家咨询委员会副主任/秘书长，中国电子学会云计算专家委员会云存储组组长；研究方向为信息网格、云计算；已主持完成基金项目18项；已发表论文80余篇，出版专著12部。

DOI: 10.3969/j.issn.1009-6868.2013.04.004
网络出版地址：http://www.cnki.net/kcms/detail/34.1228.TN.20130627.1506.004.html

超低功耗云存储系统——cStor

cStor: A Super-Low Power Consuming Cloud Storage System

中图分类号：TN915.03; TP393.03　文献标志码：A　文章编号：1009-6868 (2013) 04-0022-003

袁高峰/YUAN Gaofeng
吴亚洲/WU Yazhou
薛妍妍/XUE Yanyan
（南京云创存储科技有限公司，江苏 南京，210014）
(Nanjing Innovative Cloud Storage Technology Co., Ltd., Nanjing 210014, China)

摘要： 低功耗cStor云存储系统是一种软件与硬件相结合的系统，利用低功耗硬件设备，通过软件进行存储资源的管理，确保系统的高可靠和高可用性，有效地解决了存储系统功耗和成本问题。cStor系统中，基于ARM芯片构建的存储设备主板功耗低于5 W，单节点最大可支持16块SATA磁盘；通过cStor云存储软件管理的低功耗存储系统，在标准42U机柜中，最大可以支持1 152 TB存储容量，功耗仅为3 400 W。

关键词： 云存储；超低功耗；拍字节级

Abstract: The low-power-consuming cStor cloud storage system comprises software and hardware. Software is used to manage storage hardware, and even though this hardware consumes little power, it is still reliable and highly available. With cStor, the cost of storage power can be reduced. The motherboards of the storage devices, which are based on ARM chips, consume less than 5 W, and a single node can support up to 16 SATA HDDs. In the standard 42U rack, cStor can support up to 1152 TB storage capacity and consume only 3400 W of power.

Keywords: cloud storage; ultra low-power consumption; petabytes level

2004年，全球共有30 EB的数据；2005年跃升到50 EB；2006年达到161 EB；到2011年，已经达到2 529 EB。

这些海量信息的存取给存储技术提出了新的挑战和更高的要求。

首先，存储的数据量秩序增长不仅要求存储系统拥有大容量的存储空间，也要求存储系统有较高的可扩展性。其次，越来越多重要的数据被存储在系统里，这就要求系统有较高的可靠性、容错性、安全性。同时，对系统所占的体积也提出了更高的要求：应当尽可能的节省空间。这就要求系统存储密度高，即特定空间内所容纳的存储盘要多，这也就意味着硬件设备功耗要尽可能的低，这样才允许密集布置。功耗低的同时也能节约能源，达到降低成本的目标。

目前影响较大的集群存储系统有Google的Google File System[1]，Hadoop的HDFS[2]，Cluster File System的Lustre[3]以及RedHat的Global File System。

P2P[4]存储，也即对等存储，是指存储节点以对等模式组成的一个存储网络。现有的P2P分布式存储系统中比较出名的是MIT的CFS[5]、Berkeley的OceanStore[6]及其原型Pond[7]、微软研究院的BitVault[8]、UCSD的Total Recall[9]、清华大学的Granary[10]、北京大学的UPStore[11]。

本文中采用的低功耗cStor云存储系统是一种软件与硬件相结合的系统，系统采用双机备份容错的方式，保证不间断服务，同时软硬件高度容错，可靠性高；采用超低功耗存储服务器节点，系统存储密度高（一个标准机柜超过1 PB容量），节约能源的同时进一步降低了成本；系统可以任意增加或减少节点，可扩展性能很好；采用控制流与数据流分离的技术，对每个存储节点上数据并行读写，存储节点数目越多，整个系统的吞吐量和IO性能将呈线性增长。

1 cStor外部结构

42U超低功耗云存储系统由18台2U ARM超低功耗存储服务器节点、2台元服务器（1主1备）、2台交换机（1主1备）、机架套件组成。一般机柜只有一面可以插硬盘。cStor采用的是ARM架构，热量很小，可以双面插盘，最大可支持384块盘。cStor选用3 T硬盘，所以存储容量为1 152 TB（3×384），即1.152 PB。

整个系统的功耗包括384块5 W硬盘，24块15 W主板，2个150 W的交换机，2个200 W的Master节点，32个机箱，每个机箱4个3 W的风扇。总功耗为3 364 W。

系统具有具有超低功耗、超低价格、超高容量、高吞吐量等特点，并通

收稿日期：2013-04-16
网络出版时间：2013-06-27
基金项目：国家自然科学基金（61071121）

过软件实现对ARM超低功耗存储服务器存储空间资源进行虚拟化整合，实现软硬件故障高度容错。硬盘、主板，电源、交换机、Master服务器之间相互冗余，任何单节点出现故障，都不会影响整个系统的运行。

提供标准接口：与Google、Amazon云存储系统不同在于，本系统提供符合可移植操作系统接口（POSIX）规范的访问接口，无论是哪种系统下的应用程序，都可以不经修改就将本系统当成自己的硬盘来使用。同时，也提供专用的应用编程接口（API）接口。

2 cStor云存储系统软件

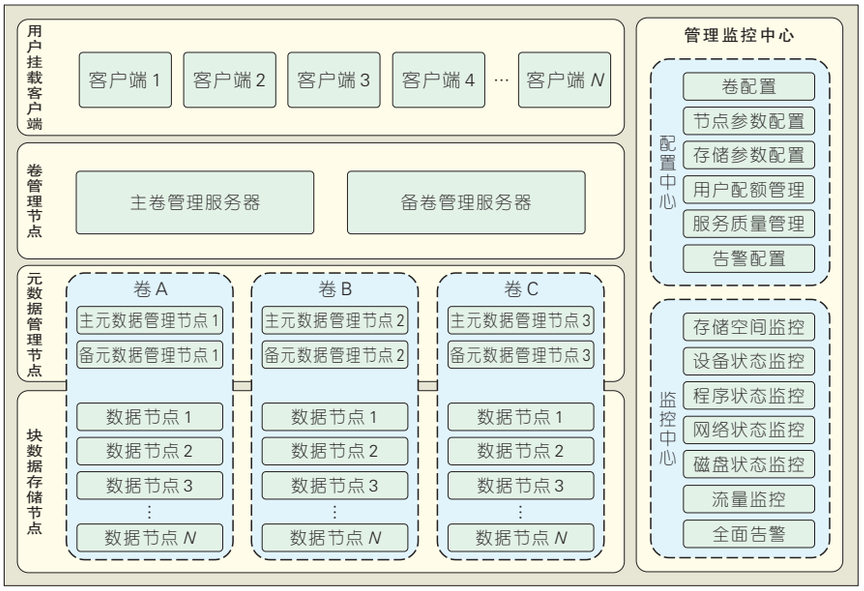
cStor云存储文件系统采用分布式的存储机制，将数据分散存储在多台独立的存储服务器上。它采用包括元数据管理服务器和数据存储节点服务器以及客户端节点的结构构成一个虚拟的海量存储卷。cStor云存储系统软件架构如图1所示。

其中，元数据管理服务器保存系统的元数据，负责对整个文件系统的管理，元数据管理服务器在逻辑上只有一个，采用主备双机镜像的方式，保证系统的不间断服务；数据存储节点服务器负责具体的数据存储工作，数据以文件的形式存储在数据存储节点服务器上，数据存储节点服务器的个数可以有多个，它的数目直接决定了cStor云存储系统的规模；客户端即为服务器对外提供数据存储和访问服务的窗口，通常客户端部署在数据存储节点服务器上，每一个块数据服务器，及时存储服务器也是客户端服务器。

使用这种系统有利于存储系统的扩展和实现，在小规模的数据扩展时，只需要添加具体的数据存储节点服务器，不需要添加整套设备。

2.1 负载自动均衡技术

cStor采用中心服务器模式来管理整个云存储文件系统，所有元数据均保存在管理节点上，文件则划分为多个块存储在不同的存储节点上。



▲图1 cStor云存储系统软件架构

管理节点维护了一个统一的命名空间，同时掌握整个系统内存储节点的使用情况，当客户端向元数据服务器发送数据读写的请求时，元数据服务器根据存储节点的磁盘使用情况、网络负担等情况，选择负担最轻的存储节点对外提供服务，自动均衡负载负担。

另外，当有一个存储节点因为机器故障或者其他原因造成离线时，管理节点会将此机器自动屏蔽掉，不再将此存储节点提供给客户端使用，同时存储在此存储节点上的数据也会自动备份到其他可用存储节点，自动屏蔽存储节点故障对系统的影响。

2.2 高速并发访问技术

客户端在访问cStor时，首先访问管理节点，获取将要与之进行交互的存储节点信息，然后直接访问这些存储节点完成数据存取。cStor的这种设计方法实现了控制流和数据流的分离。

客户端与管理节点之间只有控制流，而无数据流，这样就极大地降低了管理节点的负载，使之不成为系统性能的一个“瓶颈”。客户端与存储节点之间直接传输数据流，同时由于文件被分成多个数据块（Chunk）进行分布式存储，客户端可同时访问多个存储节点，从而使整个系统的I/O高度并行，系统整体性能得到提高。

通常情况下，系统的整体吞吐率与存储节点的数量呈正相关。

2.3 高可靠性保证技术

对于元数据，cStor通过操作日志来提供容错功能，当管理节点发生故障时，在磁盘数据保存完好的情况下，可以迅速恢复以上元数据。为了防止管理节点彻底死机的情况，cStor还提供了管理节点远程的实时备份，这样在当前的管理节点出现故障无法工作的时候，另外一台备管理节点可以迅速接替其工作。

对于存储节点，cStor采用副本的方式实现容错。每一个块有多个存储副本（默认为两个），分布存储在不同的存储节点上。副本的分布策略考虑了多种因素，如网络的拓扑、机架的分布、磁盘的利用率等。对于每一个存储节点，必须将所有的副本全部写入成功，才视为成功写入。在其后的过程中，如果相关的副本出现丢失或不可恢复等状况，管理节点会自

动将该副本复制到其他存储节点，从而确保副本保持一定的个数。在有多个存储节点的情况下，任意损失一个节点，数据都不会丢失，而且随着存储节点数目的增多，整个系统的可靠性越大。

2.4 高可用技术

由于采用了低耦合的分布式架构，所有服务节点均通过网络互连，系统可以在不停服务的情况下，通过增删节点的方式伸缩系统规模。存储节点和元数据管理服务节点间通过注册管理机制自适应管理，实现自动伸缩。

3 低功耗存储节点

针对云存储系统应用开发的低功耗主板，采用基于ARM v7架构的MV78460四核CPU，该CPU采用55 nm技术，主频1.6 GHz。采用1 GB的DDR3内存，频率为1 066 MHz。相应的组件包含有：2个10M/100M/1000M自适应网口，采用低功耗Marvell PHY 88E1318；4个SATA PM，采用4个miniSAS接口支持16块硬盘；1个2 GB NAND FLASH，用于存放内核引导程序（Bootloader）、操作系统内核（Kernel）和根文件系统（Fs）。

MS316是基于Marvell的低功耗、高端嵌入式处理器的CPU——MV78460开发的存储节点的主板。

MV78460是Marvell公司专为企业级云计算开发的ARMADA XP（极限性能）系列的工业级四核ARM CPU。ARMADA XP系列为了下一代的“绿色”系统，采用了超低功耗架构并整合了4个Marvell设计的1.6 GHz主频的ARM V7核CPU，其自带了一个IO外围设备控制器，以提供行业最强劲的表现。因为采用了高级的设计技术和制作工艺，ARMADA XP将他的云计算应用范围从高性能网络服务器渗透到了高容量服务器比如网络连接式存储（NAS）和媒体服务器。

MV78460拥有4个ARM核，它的2级高速缓存容量为2 MB，DRAM接口为32位64位可选。除了内核的强劲表现外，他还带有丰富的外围设备接口：4个千兆网口、2个SATA控制器，2个PCI-e2.0 × 4接口2个PCI-e2.0 × 4接口等。

4 cStor性能测试

4.1 硬件测试

4.1.1 云存储节点规格配置

本测试采用MS316低功耗主板的存储服务器cServer A2020，单节点容量为36 TB，包含12个3 TB SATA硬盘，共计36 TB存储容量。

4.1.2 节点性能

cServer A2020是专门针对云服务器（Cloud Server）应用开发的低功耗ARM存储服务器，即可作为云存储的存储节点，也可作为独立存储服务器使用。

cServer A2020在云存储软件系统上做了充分的测试。测试结果表明，32 kB文件在不同读写比例下的IOPS均大于2 200次/s，64 kB文件在不同读写比例下的IOPS均大于2 000次/s，1 MB文件在不同读写比例下的IOPS均大于500次/s。

4.2 系统测试

针对MS316的性能，在cStor系统上进行了充分的测试，MS316基于cStor系统的测试结果如图2所示。

由图2可知，相同存储节点的情况下，随着客户端访问数的增加，客户端读写性能逐步提升。相同客户端数目的情况下，随着节点数的增多，读写性能也有一定程度的提升，尤其在2个客户端以上时，性能提升明显。当存储节点为18个，客户端为4个时，读性能可以达到400 MB/s，写性能将近500 MB/s。

5 结束语

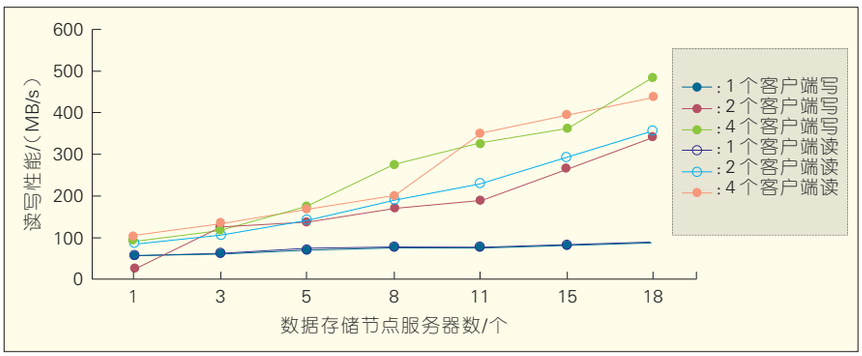
本文中的拍字节级超低功耗cStor云存储系统是一种软件与硬件相结合的系统，系统软件可以实现负载自动均衡、高速并发访问、高可靠性保证、高可用。

采用超低功耗存储服务器节点，系统存储密度高：单机架存储裸容量1.125 PB；有高度可靠的冗余备份机制；单机架总功率仅有3.4 kW，采用自主研发的超低功耗云存储硬盘节点，单节点功率仅有10 W。在标准42U机柜中，最大可以支持1 152 TB存储容量，而功耗仅为3 400 W。

采用控制流与数据流分离的技术，对每个存储节点上数据并行读写，存储节点数目越多，整个系统的吞吐量和IO性能呈线性增长。测试得知：32 kB文件在不同读写比例下的IOPS均大于2 200次/s。当存储节点为18个，客户端为4个时，读性能可以达到400 MB/s，写性能将可以达到500 MB/s。

系统中采用廉价的大容量存储

➡下转第38页



▲2 基于MS316的不同节点和不同客户端读写性能测试分析图

DOI: 10.3969/j.issn.1009-6868.2013.04.005
网络出版地址：http://www.cnki.net/kcms/detail/34.1228.TN.20130627.1503.003.html

实时云计算数据库——数据立方

DataCube: A Real-Time Cloud Computing Database

中图分类号：TN915.03; TP393.03　文献标志码：A　文章编号：1009-6868 (2013) 04-0025-007

王磊/WANG Lei
张真/ZHANG Zhen
王胤然/WANG Yinran
（南京云创存储科技有限公司，江苏 南京，210014）
(Nanjing Innovative Cloud Storage Technology Co., Ltd., Nanjing 210014, China)

摘要： 基于快速发展的并行数据库技术、云计算 MapReduce 技术及其混合技术，分析了这些技术的优缺点，对并行计算架构、分布式存储系统之上的索引以及其他方面进行了研究，提出了一种被称为数据立方的大数据处理系统。通过与大数据处理系统 Hive 和 HadoopDB 的对比实验表明，数据立方的大数据处理系统在入库、查询、并发、扩展等多方面有明显的优势。

关键词： 云计算；实时；大数据；并行计算

Abstract: In this paper, we discuss parallel database technology, MapReduce for cloud computing, and hybrid (parallel and MapReduce) technology. We discuss the advantages and disadvantages of all these technologies. We discuss parallel architecture and indexing on distributed storage system. We also discuss other aspects of big-data processing technology and propose a big-data processing system called Datacube. Datacube ios shown to have advantages over Hive and HadoopDB in terms of in query, concurrency, and expansibility.

Key words: cloud computing; real-time; large-data; parallel computing

近年来，随着计算机技术的发展，各领域数据的增长越来越快。这些数据来自方方面面，从搜集天气情况的感测器、接入社交媒体网站的指令、数码图片、在线的视频资料，到网络购物的交易记录、手机的全球定位系统信号等。随着数据规模的急剧膨胀，各行业累积的数据量越来越巨大，数据类型也越来越多、越来越复杂，已经超越了传统数据管理系统、处理模式的能力范围，传统的串行数据库系统已经难以适应这种飞速增长的应用需求。在这种需求的驱动下，云计算中的 MapReduce[1] 技术、并行数据库技术以及云计算与数据库相结合的技术应运而生。

收稿日期：2013-04-15
网络出版时间：2013-06-27
基金项目：国家自然科学基金（61071121）

本文在大数据的背景下，对大数据处理技术进行了探讨，将其分为三类：MapReduce 技术、并行数据库技术和云计算与数据库相结合的技术。通过研究这些技术的架构、适用环境，本文提出了一种全新的云计算数据库——数据立方。

1 云计算相关技术

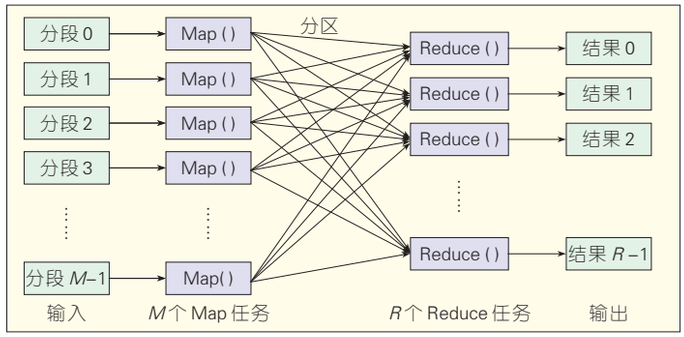
1.1 大数据处理技术——MapReduce

MapReduce 计算架构把运行在大规模集群上的并行计算过程简单抽象为两个函数：Map 和 Reduce，也就是分解与规约。简单地说，MapReduce 就是“任务的分解与结果的汇总”。程序将大数据分解为多个数据块由 Map 函数处理，Reduce 把分解后多任务处理产生的中间结果汇总起来，得到最终结果。适合 MapReduce 处理的任务特征为：待处理的大规模数据集可以切分为多个小的数据集，并且每一个小数据集都可以完全并行地进行处理。

图 1 介绍了用 MapReduce 处理大数据集的过程。一个 MapReduce 操作可以分为两个阶段：Map 阶段和 Reduce 阶段。

在映射阶段，MapReduce 并行计算架构将用户的输入数据切分为 M 个数据段，每个数据段对应 1 个 Map 任务。每一个 Map 函数的输入是数据段中的键值对<K1,V1>集合，Map 函数是用户继承 MapReduce 并行计算架构而编写的，Map 操作调用此函数，输出一组中间结果，即键值对<K2,V2>集合。接下来，按照中间结果集合的 K2 将中间结果集进行排序，生成一个新的<K2,list（V2）>集合，使得对应同一个 K2 的所有值的数据都聚集在一起。然后，按照 K2 的范围将这些元组分割为 R 个片断，对应 Reduce 任务的数目。在规约阶段，每一个 Reduce 操作的输入是一个<K2,list（V2）>片断，Reduce 操作调用用户定义的 Reduce 函数，生成用户需要的键值对<K3,V3>进行输出。

这种简洁的并行计算模型在系统层面解决了可用性、扩展性、容错



◀图 1 MapReduce 处理大数据集的过程

性等问题，是非关系数据管理和分析技术的典型代表。MapReduce 是面向廉价计算机组成的大规模集群设计的，其非共享结构、松耦合性和较强的容错能力带来了较强的扩展能力，同时，MapReduce 在工业界被广泛应用，Google、twitter、Facebook、Yahoo 等厂商对其进行了深度的改进和扩展。此外，MapReduce 的<key,value>存储模型能够存储任意格式的数据，Map 和 Reduce 函数可以进行各种复杂的数据处理，这也使得程序员的负担加重，在对上层业务的开发效率上不如结构化查询语言（SQL）简单。在相同的硬件条件下，对于有具体条件的查询来说，并行数据库[2]的性能是远远超过 MapReduce 的，但是对于在大数据上的复杂统计业务来说，MapReduce 在速度上会占有一定优势，MapReduce 是为非结构化大数据的复杂处理而设计的，这些业务具有一次性处理的特点，此外由于采取了全数据扫描的模式以及对中间结果逐步汇总的策略，使其在拥有良好扩展能力和容错能力的同时也导致了较高的磁盘和网络 I/O 的负载以及较高的数据解析代价[3]。

1.2 并行数据库技术

在 20 世纪 80 年代，数据库流行的同时并行数据库也开始起源，早期并行数据库（如 Gamma[4]和 Grace[5]）的基础架构被沿用至今，当前的并行数据库主要有 Oracle 的 Exdata[6]、EMC 的 Greenplum[7]、Teradata[8]，这些数据库都支持标准 SQL。并行数据库一般可以分为无共享架构（Shared-nothing）和磁盘共享存储架构（Shared-disk）两种存储架构，如图 2 所示。这两种架构有各自的优缺点，在 Shared-nothing 系统中，数据集被切分成为了多个子集[9-11]，集群中每个节点分别存储一个子集在本地磁盘上，一般来说，Shared-nothing 系统可以提供很高的并行 I/O 和并行计算能力，但是也有多节点事务处理[12-13]、数据传输以及数据倾斜[14]等问题。在 Shared-disk 系统中，数据被集中存储，所有的数据库节点都可以访问存储系统的任意一个磁盘，因此数据也没有必要被切分，这也避免了数据倾斜的问题，这种系统主要的缺陷在于较低的 I/O 带宽和扩展能力。

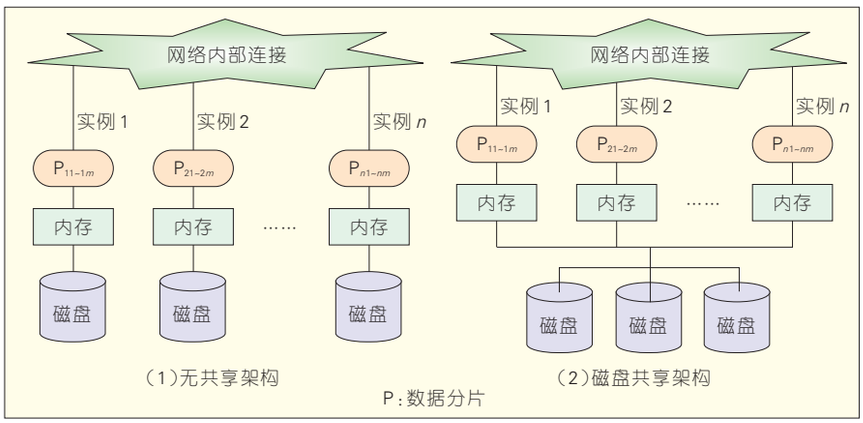
1.3 云计算与数据库相结合的技术

与数据库相结合的云计算技术一般指的是 MapReduce 技术，当前主要有 Teradata 公司的 Aster Data[15]和耶鲁大学提出的 HadoopDB[16]。

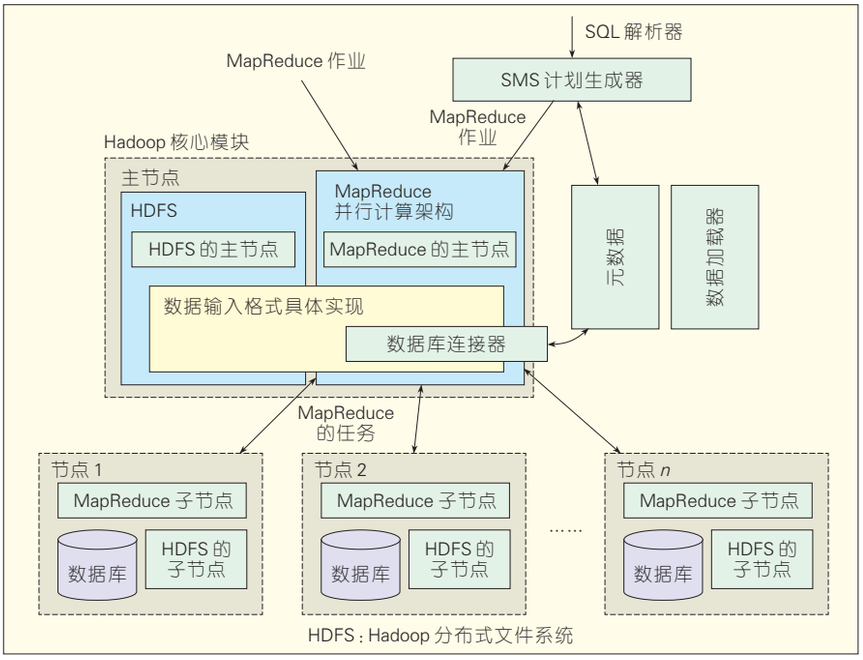
Aster Data 将 MapReduce 与 SQL 引擎相结合，针对大数据处理和分析提出了 SQL/MapReduce 框架，用户可以使用 JAVA、C++等多种语言在 Aster Data 的并行框架上编写 MapReduce 函数，编写的函数可以作为一个子查询在 SQL 中使用，从而获得 SQL 的易用性和 MapReduce 的开放性。同时 Aster Data 能够对多结构化数据、原始数据进行处理和分析，并拥有丰富的统计软件包可以讲数据分析推向数据库内进行，提升了数据分析性能。

在 HadoopDB 中，系统清晰地分成两层，上层使用 Hadoop 进行任务的分解和调度，下层用 RDBMS（Postgresql）进行数据的查询和处理，在处理查询时，执行的是 SQL to mapReduce to SQL 操作过程（SMS planner）。该工作的创新之处是：试图利用 Hadoop 的任务调度机制提高系统的扩展性和容错性，以解决大数据分析的横向扩展问题；利用 RDBMS 实现数据存储和查询处理，以解决性能问题。在其性能实验中，HadoopDB 的性能仍然落后于关系数据库系统。如何提升 MapReduce 的性能，已引起研究人员的高度重视，研究人员提出了 MapReduce 的各种优化技术，获得了重要的性能改进。Yale 大学 Abadi 领导的小组正在使用包括列存储、持续装载和分析等技术，以改进 HadoopDB 的性能[17]。

图 3 所示是 HadoopDB 的一个结



▲图 2 无共享架构（Shared-nothing）和磁盘共享存储架构（Shared-disk）



▲图 3 HadoopDB 结构图

构图，在原来的 Hadoop 与 hive 的基础上，增加了一些组件[18]：其中 SMS Planner 的作用是在 hive 解析 SQL 语句生成 MapReduce 任务树之后，对 MapReduce 任务树进行优化，指导 Hadoop 去并行数据库中执行 SQL。Catalog 里面存储了并行数据库的一些信息。Data loader 负责把原始数据加载到并行数据库中，需要完成的工作是对原始数据的划分。Database Connector 用于向各个节点传递信息，包含了节点里面数据库的链接信息和需要执行的 SQL 语句。Paralled DataBase 用于代替 HDFS 在各个节点上存储数据。

2 云计算数据库——数据立方

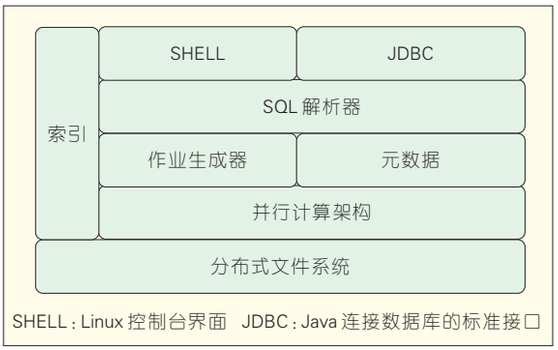
通过对 MapReduce、并行数据库和两者的混合技术研究，南京云创存储科技有限公司推出了实施云计算数据库——数据立方，该系统通过引入索引模块、并行执行架构以及读取本地磁盘的执行方式，使查询达到了实时完成、简单易用、高可靠安全的效能，使艾字节级的数据能够秒级处理，极大地提高了用户执行查询操作后的使用效率，不仅在查询和检索这部分数据的时候具有非常高的性能优势，数据立方还可以支持数据仓库存储、数据深度挖掘和商业智能分析等业务。

2.1 数据立方的体系架构

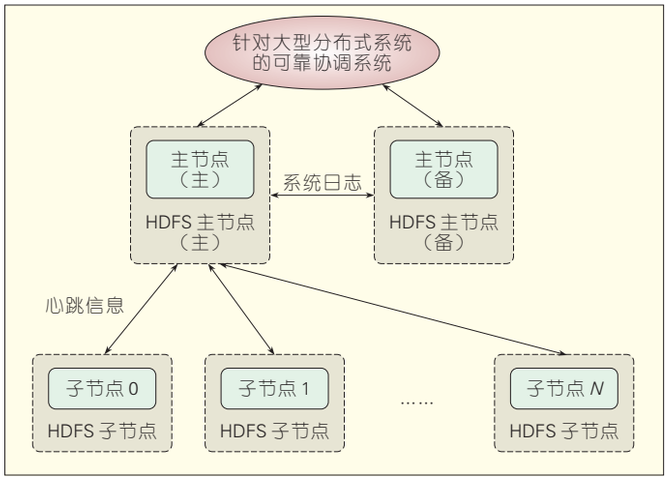
数据立方（DataCube）的结构分为用户接口、索引、SQL 解析器、作业生成器、元数据管理、并行计算架构、分布式文件系统等部分，如图 4 所示。用户接口主要有两个：JDBC 和 Shell。JDBC 主要执行数据的定义操作，即建立数据库、建表、建分区，对数据库、表和分区的删改等，同时可执行数据查询的 SQL 语句，暂不支持单条记录的增删改；数据立方提供友好的 Shell 交互界面，Shell 支持数据库、表的增删改以及数据查询的 SQL 语句。数据在入库的同时与数据对应的索引也在同时建立，索引是一颗 B 树，数据插入到内存的同时，索引 B 树也在生成，当达到设置上限时，数据和索引会刷新到分布式文件系统上成为文件。数据立方的元数据存储在数据库中。其中包括，数据库的名字和属性，数据库中的表，表的名字，表的列和分区及其属性，表的属性，表的数据所在目录等等。SQL 解析器接收从 JDBC 和 Shell 传来的 SQL 查询语句，同时对 SQL 进行词法分析、语法分析、编译、优化。作业生成器根据 SQL 语法树生成查询作业，分析所要处理的数据表对应的索引文件的所在存储子节点位置，并将作业发送给并行计算架构。并行计算架构接收到作业生成器生成的作业，根据索引文件的位置切分查询作业形成子任务，然后将子任务发送给数据所在的存储子节点，每个节点执行这些子任务查询索引得到结果记录所在的数据文件名与偏移量，并以广播的方式发送查询子任务到数据文件所在的节点，在执行完毕后将结果返回。数据立方可以使用 HDFS 和 cStor[19] 作为底层存储系统，cStor 是一个主从结构的分布式文件系统，不仅具有 HDFS 的高吞吐率、高读写性能等特性，还支持 HDFS 所不具备的对文件修改等功能，并且支持 POXIS 接口。

2.2 分布式并行计算架构（DPCA）

数据立方的分布式并行架构（DPCA）是典型的主从结构，如图 5 所示。主 Master 与从 Master 分别部署在 HDFS 的主从 NameNode 物理节点



▲图 4 数据立方架构



◀图 5 DPCA 架构

上，而 Slave 部署在 DataNode 物理节点上，主从 Master 使用 Zookeeper 同步，并共享系统日志，Master 与 Slave 之间用心跳信息保持信息交换。

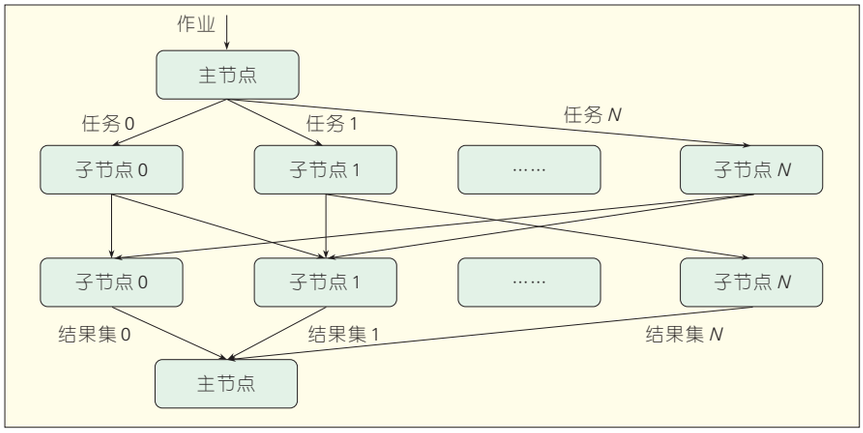
相对于 MapReduce 架构，DPCA 具有实时性、计算的数据本地性以及数据平衡性。MapReduce 架构的作业（Job）提交过程较为复杂，客户端将 Job 提交到 JobTracker 有较长的延迟，JobTracker 将 Job 处理为 MapReduce Task 后，通过 TaskTracker 的心跳信息将 Task 任务返回给 TaskTracker，此过程中也存在延迟。

MapReduce 架构虽然也遵循数据本地性，但仍会有很大比例的数据处理不是本地的，相对于 MapReduce 架构，DPCA 的 Job 提交是实时性的，在提交 Job 之前所需程序 Jar 包已经分发到所有计算节点，在 Job 提交之后，Master 在初始化处理之后即将 Task 直接分发到所有 Slave 节点上，如图 6 所示，在 Job 提交后，Master 根据数据文件所在位置分配 Task，这样在每个计算节点上要处理的 HDFS 上的数据块就在本地，这样避免了数据的移动，极大地减少了网络 IO 负载，缩短了计算时间，每个计算节点会根据 Task 中 SQL 解析器生成的执行计划对 Task 执行的结果进行分发，分发的方式有 3 种：分发所有中间数据到所有计算节点、分发所有中间数据到部分节点、根据数据所在位置分发，如图 7 所示。并行计算架构能够周期性地对 HDFS 上的数据表进行维护，保持数据表在所有的 DataNode 节点上所存储的数据量的平衡，减少因数据负载的不平衡而导致的计算负载的不平衡。

举一个典型的小表与大表 Join 连接的实例，如图 8 所示，Master 解析 Job 中的执行计划，判断小表的位置后，将 Task0 发送给了 Slave0，指令 Slave0 发送小表到所有节点，而其他节点接收到的子任务是等待接受小表的数据，接收到数据后将小表与大表连接并将数据返回给 Master，当所有数据返回完成则这个 Job 完成。

2.3 分布式索引

MapReduce 是对每个查询都是直



▲图 6 并行计算架构上作业执行过程

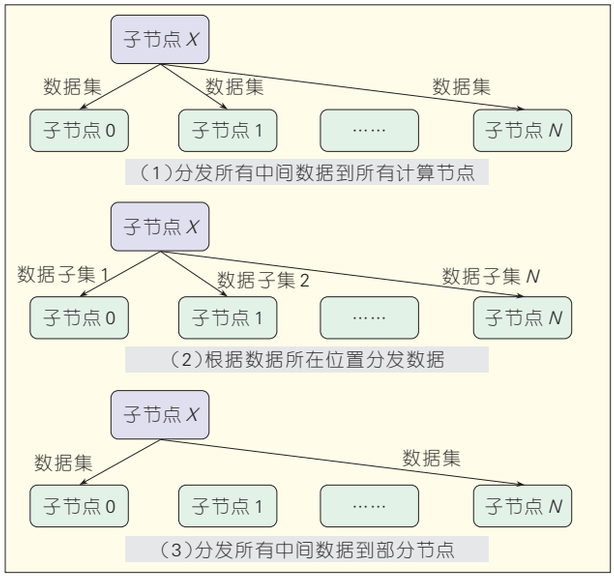
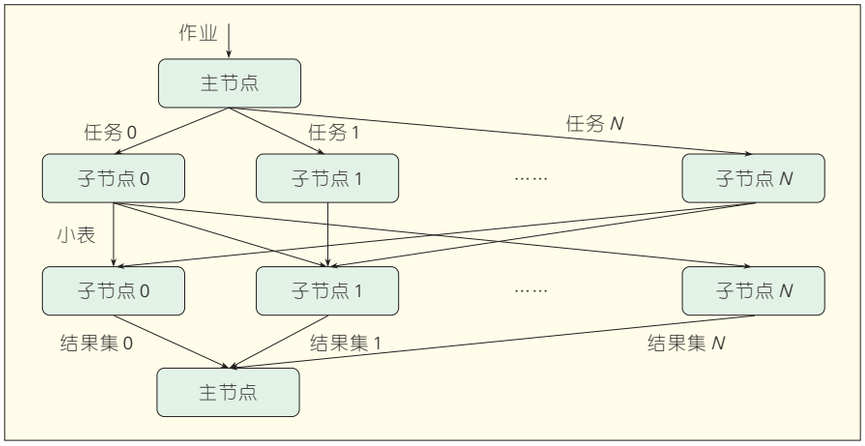


图 7▶ 并行计算架构的 3 种分发方式



▲图 8 小表与大表的 Join 实例

接从分布式文件系统中读入原始数据文件，I/O 代价远高于数据库，相对于 MapReduce 架构以及在其之上的 SQL 解析器 Hive，数据立方引入了一种高效的分布式索引机制，不同于并行数据库的 Shared-nothing 和 Shared-disk 架构，数据立方的数据文件与索引文件都存放在分布式文件系统之上。

数据在入库的同时 B 树索引在内存中同步生成，B 树中的叶子节点存储的是数据文件路径与记录在文件中的偏移量，如图 9 所示，在 B 树中的叶子节点达到设置上限后，索引将被序列化到分布式文件系统之上，在根据条件进行单表查询的时，Job 被提交到并行计算框架，Master 节点首先分析该表的索引文件根据索引文件所在的节点将 Task 发送到相应的节点，每个节点在查询本地的索引文件之后将符合条件的数据文件路径+偏移量打包成 Task 根据数据文件位置进行再次分发，在数据文件中的记录查询出来之后将结果返回，如图 9 所示。

3 实验与评估

3.1 实验环境

实验环境搭建在两个机架的 12 台物理机组成的集群上。每台物理机使用 Ubuntu9.04 server 系统，JDK 版本为 1.6.0.18，使用的 Hadoop 版本为 2.0.0，将 HDFS 作为分布式存储环境。软硬件配置如表 1、表 2 所示。

当前与数据立方类似的产品有分布式数据库和数据仓库，如：开源的 HIVE、HadoopDB 等，因此我们在数据入库、查询、查询的并发量以及线性扩展等多方面对数据立方、HIVE 和 HadoopDB 做了对比实验。

3.2 数据入库实验

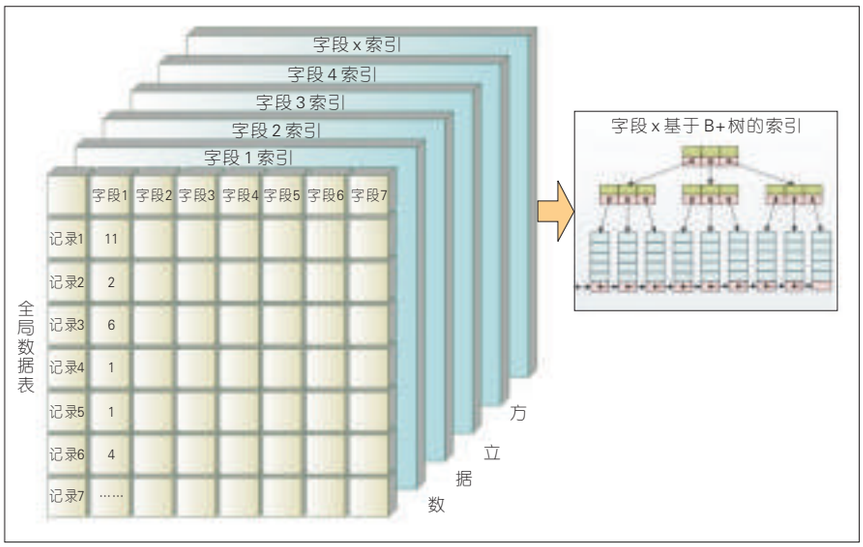
数据立方能够快速进行数据入库同时实时建立索引，相对于基于传统数据库的 HadoopDB 来说具有天然的优势，但由于 HIVE 在数据入库的同时并没有建立索引使其在查询的过程中没有优势。实验结果如图 10 所示。

3.3 单表查询实验

对于简单的单表查询来说，数据量较小时，HadoopDB 与数据立方的查询速度都是比较快的，但在大数据量下，数据立方的高效分布式查询更有优势，而 HIVE 的底层是基于 MapReduce，所以速度较慢。实验结果如下图 11 所示。

3.4 多表查询实验

在多表查询方面，在小表与小



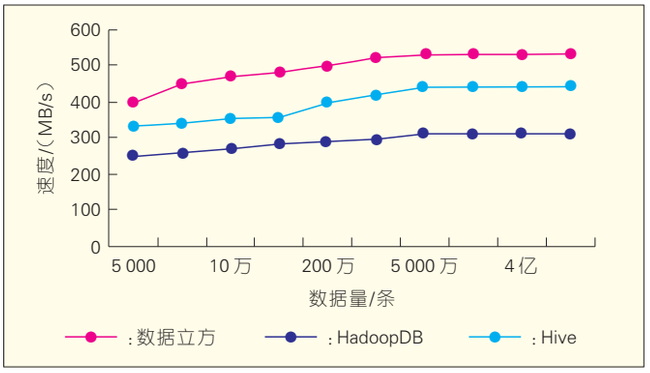
▲图 9 B 树索引

▼表 1 硬件配置

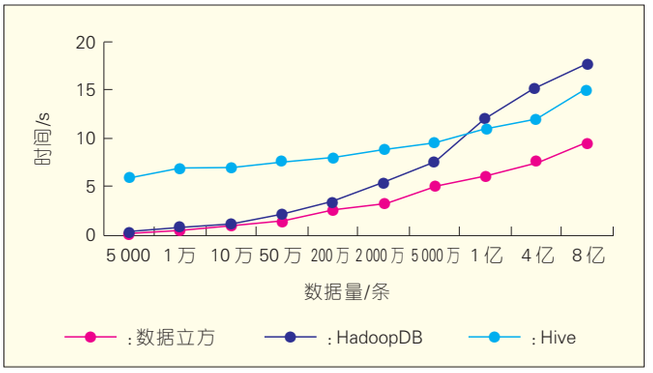
设备名称	数量	CPU	内存	硬盘
主控制服务器	2	双路四核，主频 2GHz	32G	2T×8
子处理服务器	10	双路四核，主频 2GHz	32G	2T×8
客户端	5	单路双核，主频 2GHz	8G	1T
48 口千兆交换机	1	—	—	—

▼表 2 软件配置

软件名称	软件版本
CentOS	6.3
HadoopDB	0.1.1.0
Hive	0.9.0
数据立方	1.0
Hadoop	2.0.0



▲图 10 数据入库实验



▲图 11 单表查询实验

表、大表与小表之间的关联查询，数据立方和 HadoopDB 都是较快的，但在大表与大表之间做关联查询时，数据立方相对于 HadoopDB 更快，而 HIVE 是最慢的。多表查询实验结果如图 12 所示。

3.5 并发查询实验

数据立方的每个节点支持 200 个并发查询，同时每个查询均是秒级响应，HadoopDB 由于是 SMS 的中间层，由于 MapReduce 架构本身的心跳机制而导致了较大的延迟，所以是很难达到秒级响应的，HIVE 的任务并发数取决于 MapReduce 的并发任务数，所以会更低。实验结果如图 13 所示。

3.6 线性扩展实验

数据立方、HadoopDB 和 HIVE 均支持线性扩展，而数据立方的扩展效率更高，即对系统的软硬件做扩展后，性能也能够达到类似线性的增长。实验结果如图 14 所示。

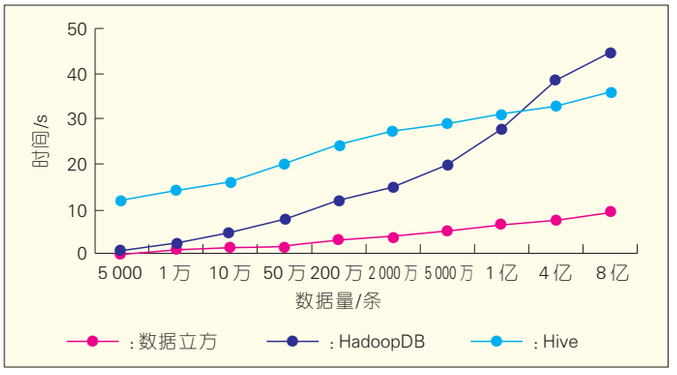


图 12▶ 多表查询实验

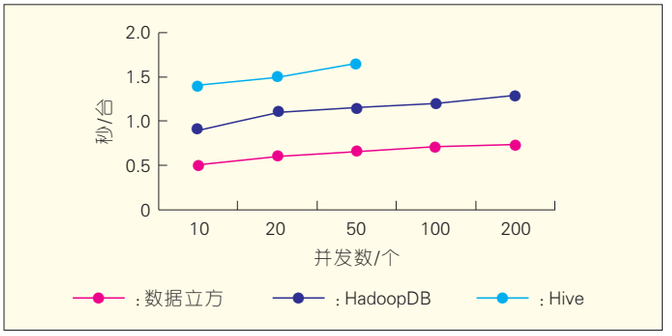


图 13▶ 并发查询实验

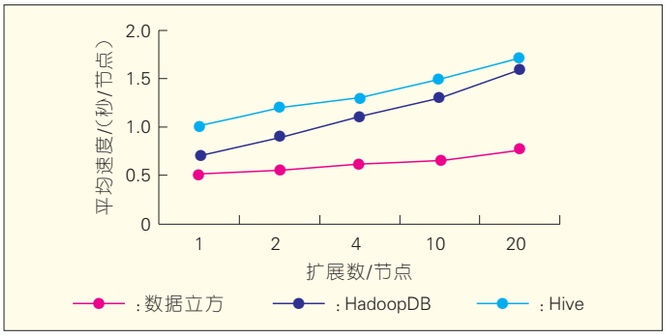


图 14▶ 线性扩展实验

4 结束语

Hadoop 是一种流行的 MapReduce 计算模型的开源实现，用于大规模数据集的并行化分析处理，并行数据库是在单机数据库基础之上发展而来的数据库集群，本文通过研究 MapReduce 技术、并行数据库技术以及混合技术探讨了一系列相关的大数据处理技术，更深一步探索了基于分布式文件系统的并行计算架构和分布式海量数据实时索引机制，以此为基础并辅以其他技术形成了一个支持非结构化、结构化和半结构化数据高效存储，支持离线数据分析和在线专题应用，支持结构化数据与非结构化、半结构化数据之间的复杂计算

的实时云计算数据库数据立方。最后，本文通过实验验证了数据立方相对于其他系统的优势。

参考文献

[1] DEAN J, GHEMAWAT S. MapReduce: Simplified data processing on large clusters [C]//Proceedings of the 6th USENIX Symposium on Operation Systems Design and Implementation (OSDI'04), Dec 6–8, 2004, San Francisco, CA USA. New York, NY, USA: ACM, 2004:137–150.
[2] PAVLO A, PAULSON E, RASIN A, et al. A comparison of approaches to large scale data analysis [C]//Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD'09), Jun 29–Jul 2, 2009,Providence,Rhode Island, USA. New York, NY, USA: ACM, 2009: 165–178.
[3] JIANG D, OOI B C, SHI L, et al. The performance of MapReduce: An in-depth study [J]. Proceedings of the VLDB Endowment (PVLDB), 2010, 3 (1): 472–483.
[4] DEWITT D J, GERBER R H, GRAEFE G, et al. GAMMA– A high performance dataflow database machine [C]//Proceedings of the 12th International Conference on Very Large Data Bases (VLDB'86), Aug 15–18, 1986, Kyoto, Japan. San Francisco, CA,USA: Morgan Kaufmann Publishers, 1986: 228–237.
[5] FUSHIMI S, KITSUREGAWA M, TANAKA H. An overview of the system software of a parallel relational database machine [C]// Proceedings of the 12th International Conference on Very Large Data Bases(VLDB'86), Aug 15–18, 1986, Kyoto,Japan. San Francisco, CA,USA: Morgan Kaufmann Publishers, 1986:209–219.
[6] EMC Corporation. Greenplum [EB/OL]. [2013–04–02]. http://www. greenplum.com/.
[7] Oracle Exadata [EB/OL]. [2013–04–09]. http://www.oracle.com/cn/products/database/exadata/overview/index.html/.
[8] Teradata Corporation. Teradata [EB/OL]. [2013–04–10]. http://www.teradata.com/.
[9] DEWITT D, GRAY J. Parallel database systems: The future of high performance database systems [J]. Communications of the ACM, 1992,35(6):85–98.
[10] MEHTA M, DEWITT D J. Data placement in Shared-nothing parallel database systems [J]. The VLDB Journal, 1997,6(1):53–72.
[11] CHAMBERLIN D D, SCHMUCK F B. Dynamic data distribution(D3) in a Shared-nothing multiprocessor data store [C]//Proceedings of the 18th International Conference on Very Large Data Bases(VLDB'92), Aug 23–27, 1992,Vancouver, Canada. San Francisco, CA, USA: Morgan Kaufmann Publishers, 1992:163–174.
[12] MAREK R, RAHM E. Performance evaluation of parallel transaction processing in Shared nothing database systems [C]// Proceedings of the 4th International Conference on Parallel Architectures and Languages Europe(PARLE'92), Jun 15–18, 1992, Paris, France. Berlin, Germany: Springer-Verlag,1992:295–310.
[13] JENQ B C, TWICHELL B C, KELLER T W. Locking performance in a Shared nothing parallel database machine [J]. IEEE Transactions on Knowledge and Data Engineering, 1989,1(4): 530–543.
[14] LEE C, CHANG Z A. Workload balance and page access scheduling for parallel JOINs in Shared-nothing systems [C]//Proceedings of the 9th International Conference on Data Engineering, Apr 19–23, 1993, Vienna, Austria. Washington, DC, USA: IEEE Computer Society, 1993:411–418.
[15] Asterdata Corporation. Asterdata [EB/OL]. [2013–04–10]. http://www. asterdata.com/.
[16] ABOUZEID A, BAJDA-PAWLIKOWSKI K, ABADI D J, et al. HadoopDB: An architectural hybrid of MapReduce and DBMS technologies for analytical workloads [C]//Proceedings of the 35th International Conference on Very Large Data Bases (VLDB'09), Lyon,France.2009: 733–743.
[17] ABOUZIED A, BAJDA-PAWLIKOWSKI K, HUANG J W, et al. HadoopDB in action: Building real world applications [C]// Proceedings of the ACM SIGMOD International Conference on Management of Data(SIGMOD'10), Jun 6–10, 2010, Indianapolis, IA, USA. New York, NY, USA: ACM, 2010:1111–1114.
[18] HadoopDB 数据仓库简介 [EB/OL]. [2013–04–10]. http://blog.csdn.net/suwei19870312/ article/details/7242995.
[19] Cstor Corporation. cstor [EB/OL]. [2013–04–10]. http://www.cstor.cn.

作者简介

王磊，中国矿业大学计算机学院硕士毕业；南京云创存储科技有限公司技术总监；从事大数据处理、数据立方产品技术规划及架构设计，大数据处理项目需求分析等。

张真，北京科技大学 MBA 硕士毕业，南京云创存储科技有限公司董事长兼 CEO。

王胤然，南京航空航天大学毕业；南京云创存储科技有限公司云计算高级研发工程师；从事分布式数据处理和大规模数据挖掘工作。

综合信息

三网融合试点初见规模 行业迎来发展新阶段

国家新闻出版广电总局发展研究中心于7月4日在北京举行《中国广播电影电视发展报告（2013）》（广电蓝皮书）出版发布会，发布了2012年全国广播影视发展的新进展、新亮点以及2013年发展总体趋向的最新研究成果。

蓝皮书称：三网融合两批共54个试点城市基本遍布全国，覆盖人口超过3亿。目前，三网融合已取得阶段性成果，正向更高层次推进。

据专家分析称：今后三网融合将出现内容为王、融合发展和互联网主导的发展趋势。

中国工程院副院长邬贺铨估算：未来3年内，三网融合启动的相关产业市场规模巨大，将超过6 000亿元人民币，其中包括电信宽带升级、广电双向网络改造、机顶盒产业发展，以及基于音视频内容的信息服务系统建设的有效投资，初步估算达2 490亿元；可激发和释放社会的信息服务与终端消费近4 390亿元；数字内容开发制作、机顶盒生产与安装等等将可以新增就业岗位20万个。

专家还认为：未来广电系、电信系、互联网三方的力量在三网融合拓展中难度较大，三方应寻求差异化经营，以避免内耗。（转载自C114中国通信网）

DOI: 10.3969/j.issn.1009-6868.2013.04.006
网络出版地址：http://www.cnki.net/kcms/detail/34.1228.TN.20130624.1519.002.html

基于云计算的大数据挖掘平台

Big Data Mining Platform Based on Cloud Computing

中图分类号：TN915.03; TP393.03　文献标志码：A　文章编号：1009-6868 (2013) 04-0032-007

何清/HE Qing
庄福振/ZHUANG Fuzhen
（中国科学院计算技术研究所 智能信息处理重点实验室，北京，100190）
(Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)

摘要： 开发了一个基于云计算的并行分布式大数据挖掘平台——PDMiner。PDMiner实现了各种并行数据挖掘算法，如数据预处理、关联规则分析以及分类、聚类等算法。实验结果表明，并行分布式数据挖掘平台PDMiner中实现的并行算法，能够处理大规模数据集，达到太字节级；具有很好的加速比性能；实现的并行算法可以在商用机器构建的并行平台上稳定运行，整合了已有的计算资源，提高了计算资源的利用效率；可以有效地应用到实际海量数据挖掘中。在PDMiner中还开发了工作流子系统，提供友好统一的接口界面方便用户定义数据挖掘任务。

关键词： 云计算；分布式并行数据挖掘；海量数据

Abstract: In this paper, we develop a parallel and distributed data mining toolkit platform called PDMiner. This platform is based on cloud computing. PDMiner is used to preprocess data, analyze association rules, and parallel classification and clustering. Our experimental results show that the parallel algorithms in PDMiner can tackle data sets up to one terabyte. They are very efficient because they have good speedup, and they are easily extended so that they can be executed in a cluster of commodity machines. This means that full use is made of computing resources. The algorithms are also efficient for practical data mining. We also develop a knowledge flow subsystem that helps the user define a data mining task in PDMiner.

Keywords: cloud computing; parallel and distributed data mining; big data

随着物联网、移动通信、移动互联网和数据自动采集技术的飞速发展以及在各行各业的广泛应用，人类社会所拥有的数据面临着前所未有的爆炸式增长。美国互联网数据中心指出，互联网上的数据每年以50%的速度增长，每两年翻一番，而目前世界上90%以上的数据是最近几年才产生的，人类社会进入了“大数据”时代。因此，信息的获取非常重要，一定程度上，信息的拥有量已经成为决定和制约社会发展的重要因素。

数据挖掘作为信息获取的一门重要技术，得到了广泛的研究。数据挖掘[1]从大量的数据中挖掘出有用的信息，提供给决策者做决策支持，有着广阔的应用前景。由于要挖掘的信息源中的数据都是海量的，而且以指数级增长，传统的集中式串行数据挖掘方法不再是一种适当的信息获取方式。因此扩展数据挖掘算法处理大规模数据的能力，并提高运行速度和执行效率，已经成了一个不可忽视的问题。

为了解决海量数据的挖掘问题，一种简单的方式就是把所有的数据划分成若干份，也就是切分成若干个子任务，然后分布到各个计算资源上去进行计算，每个节点完成一个子任务，最后进行集成。分布式计算就是把一个计算问题分解成多个子问题并同时处理的计算模型。基于分布式计算模型，Luo等人[2-4]集成了很多数据挖掘算法到多主体系统。另外一种提高计算效率的方式是并行计算，并行计算也是把一个大的计算问题分割成小任务的形式。近年来，并行计算的体系结构和模型也引起了广泛的兴趣和研究[5-6]。

尽管分布式计算和并行计算有很相似的特点，但是它们之间各有侧重，分布式计算强调在所有异构计算资源上同时求解问题，而并行计算则更加强调同一台计算资源内部多线程并行。这两种计算方式可以对应到算法之间的并行以及算法内部并行这两种计算模式。文献[2-4]提出基于主体技术的算法之间并行的计算模式，他们利用主体技术中主体本身的自主性、智能性等特点，实现不同算法主体之间的并行计算，以消息传递的方式实现同步，大大提高了算法的执行效率，减少了运行时间。第二种计算模式，是粒度比较小的并行

收稿日期：2013-04-15
网络出版时间：2013-06-24
基金项目：国家自然科学基金（61175052、61203297）；国家高技术研究发展（“863”）计划（2013AA01A606、2012AA011003）；国家重点基础研究发展（“973”）规划（2013CB329502）

方式，主要研究的是算法内部的并行。通过把算法分解，尽可能地找出算法中可并行的部分进行并行计算。这种计算模型的最终效率取决于算法本身的可并行程度，如果并行程度非常高，那么就可以大大提高算法的运行效率。由于在很多应用中，只需要执行一种应用（算法），所以研究算法内部的并行实现非常重要。文献[7]实现了多种机器学习算法在多核计算机上的并行，本文主要针对第二种并行计算模式进行研究，而且可以在大规模计算机集群上运行。

近年来，云计算得到了学术界和业界的广泛关注，它是一种基于互联网的、大众参与的计算模式，其计算资源，包括计算能力、存储能力、交互能力，是动态、可伸缩、且被虚拟化的，以服务的方式提供给用户。基于大规模数据处理平台——Hadoop，我们研究开发了并行分布式数据挖掘平台——PDMiner，其目的是设计实现并行数据挖掘算法处理大数据集，且提高执行效率。在PDMiner中包含4个子系统，工作流子系统、用户接口子系统、数据预处理子系统和数据挖掘子系统。整个数据挖掘平台提供了一个从海量数据中挖掘有用知识的完整解决方案，而且提供了可扩展的灵活接口。

1 大规模数据处理平台——Hadoop

Hadoop是一个软件计算平台，可以让程序员很容易地开发和运行处理海量数据的应用程序。其核心部分包括HDFS[8]和基于MapReduce[9-10]机制的并行算法实现。

1.1 HDFS

Hadoop分布式文件系统HDFS是受Google文件系统启发，建立在大型集群上可靠存储大数据集的文件系统。它和现有的分布式文件系统有着很多的相似性，然而和其他的分布式文件系统的区别也是很明显的。HDFS具有高容错性，可以部署在低成本的硬件之上。此外，HDFS提供高吞吐量地对应用程序数据的访问，适合大数据集的应用程序。

HDFS结构包含一个名字节点作为控制主节点，其他的服务器作为数据节点，存储数据。具体地说，HDFS具有如下几大特点：

（1）强容错性

HDFS通过在名字节点和数据节点之间维持心跳检测、检测文件块的完整性、保持集群负载均衡等手段使得系统具有高容错性，集群里个别机器故障将不会影响到数据的使用。

（2）流式数据访问与大数据集

运行在HDFS之上的应用程序必须流式地访问它们的数据集。HDFS适合批量处理数据，典型的HDFS文件是吉字节到太字节的大小，典型的块大小是64 MB。

（3）硬件和操作系统的异构性

HDFS的跨平台能力毋庸置疑，得益于Java平台已经封装好的文件IO系统，HDFS可以在不同的操作系统和计算机上实现同样的客户端和服务端程序。

1.2 MapReduce

MapReduce是Google实验室提出的一种简化的分布式程序设计模型，用于处理和生成大量数据集。通过该模型，程序自动分布到一个由普通机器组成的超大机群上并发执行。

Map和Reduce是该模型中的两大基本操作。其中，Map是把一组数据一对一的映射为另外的一组数据，Reduce是对数据进行规约，映射规则与规约规则可由用户通过函数来分别指定。现实生活中很多任务的实现都是可以基于类似这样的映射规约模式。

MapReduce通过把对数据集的大规模操作分发给网络上的每个节点来实现可靠性，每个节点会周期性地把完成的工作和状态信息返回给主节点。如果一个节点保持沉默超过一个预设的时间间隔，主节点就认为该节点失效了，并把分配给这个节点的数据发到别的节点，并且因此可以被其他节点所调度执行。

由于MapReduce运行系统已考虑到了输入数据划分、节点失效处理、节点之间所需通信等各个细节，使得程序员可以不需要有什么并发处理或者分布式系统的经验，就可以处理超大规模的分布式系统资源。

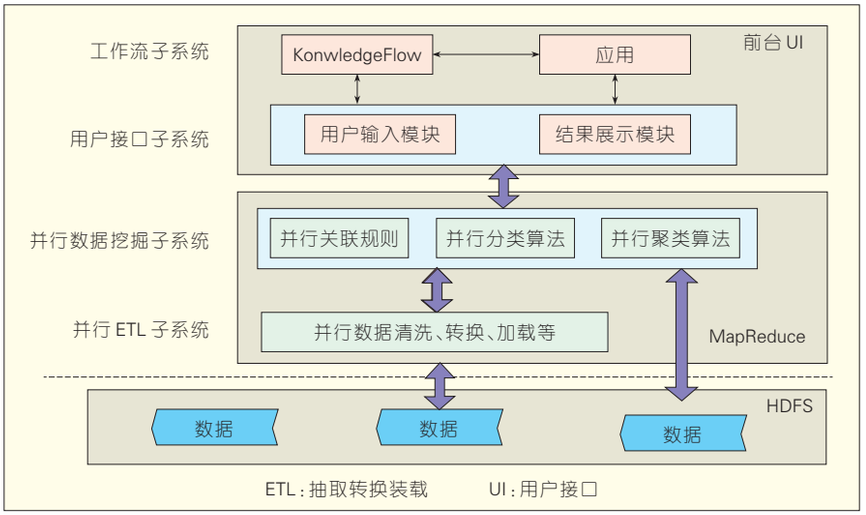
2 并行分布式大数据挖掘平台体系架构

Hadoop提供了让程序员易于开发和运行处理海量数据应用程序的平台，其分布式文件系统HDFS是建立在大型集群上可靠存储大数据集的文件系统，具有可靠性，强容错性等特点；MapReduce提供了一种高效编写并行程序的编程模式。基于此，我们开发了并行数据挖掘平台——PDMiner，大规模数据存储在HDFS上，且通过MapReduce实现各种并行数据预处理和数据挖掘算法。

PDMiner是一个集成各种并行算法的数据挖掘平台，其中的并行计算模式不仅包括算法之间的并行，而且包括算法内部的并行。图1给出了并行数据挖掘平台PDMiner的总体系统架构，其中主要包括4个子系统：工作流子系统、用户接口子系统、并行抽取转换装载（ETL）子系统以及并行数据挖掘子系统。工作流子系统提供了友好的界面方便用户定义各种数据挖掘任务；用户接口可以对算法的参数进行设置以及通过结果展示模块分析挖掘结果并做出相应的决策；并行ETL算法子系统和并行数据挖掘算法子系统是PDMiner的核心部分，它们可以直接对存储在HDFS系统上的数据进行处理，ETL算法处理后的结果也可以作为数据挖掘算法的输入。

2.1 工作流子系统

工作流子系统提供了友好和统



▲图1 并行数据挖掘平台 PDMiner 的体系结构

▲图2 并行朴素贝叶斯算法参数设置界面

一的用户接口(UI)，使得用户可以方便地建立数据挖掘任务。在创建挖掘任务过程中，可以选择 ETL 数据预处理算法、分类算法、聚类算法、以及关联规则算法等，右边下拉框可以选择服务单元的具体算法。工作流子系统通过图形化 UI 界面为用户提供服务，灵活建立符合业务应用工作流程的自定制挖掘任务。通过工作流界面，可以建立多个工作流任务，不仅每个挖掘任务内部并行，而且不同数据挖掘任务之间也并行。

2.2 用户接口子系统

用户接口子系统由 2 个模块组成：用户输入模块、结果展示模块。用户接口子系统负责与用户交互，读写参数设置，接受用户操作请求，根据接口实现结果展示。比如并行分类算法中并行朴素贝叶斯算法的参数设置界面如图 2 所示，从图中看到可以方便地设置算法的参数。这些参数包括训练数据、测试数据、输出结果以及模型文件的存储路径，而且还包括 Map 和 Reduce 任务个数的设置。结果展示部分实现了结果可视化理解，比如生成直方图、饼图等。

2.3 并行 ETL 算法子系统

数据预处理算法在数据挖掘中起着非常重要的作用，其输出通常是数据挖掘算法的输入。由于数据量的剧增，串行数据预处理过程需要消耗大量的时间来完成操作过程，因此为了提高预处理算法的执行效率，在并行 ETL 算法子系统中设计开发了 19 种预处理算法[11]，如图 3 所示，包括并行采样 Sampling、并行数据预览 PDPreview、并行数据添加标签 PDAddLabel、并行离散化 Discretize、并行增加样本 ID、并行属性交换 AttributeExchange、并行布尔型数据到系列数据的转换 BoolToSerialNum、并行数据归一化 Normalize、并行属性约简 PCA、并行数据集成 DataIntegration、并行统计 Statistic、并行属性约简 AttributeReduction、并行数据区间化 Intervalize、并行冗余数据删除 RedundancyRemove、并行属性添加 AttributeAdd、并行属性修改 AttributeModify、并行数据缺失值替换 ReplaceMissingValues、并行属性删除 AttributeDel，以及并行属性选择 AttributeSelection 等。

通常 ETL 操作都具有很高的并行化程度，比如属性的删除，可以把数据划分成很多块，算法对每个数据块的处理都是相对独立的，因此并行 ETL 子系统中实现的并行 ETL 算法具有很好的加速比，大大提高了算法的运行速度和执行效率。

2.4 并行数据挖掘子系统

并行数据挖掘子系统是并行数据挖掘平台 PDMiner 的核心部分，主要包括了三大类算法：并行关联规则

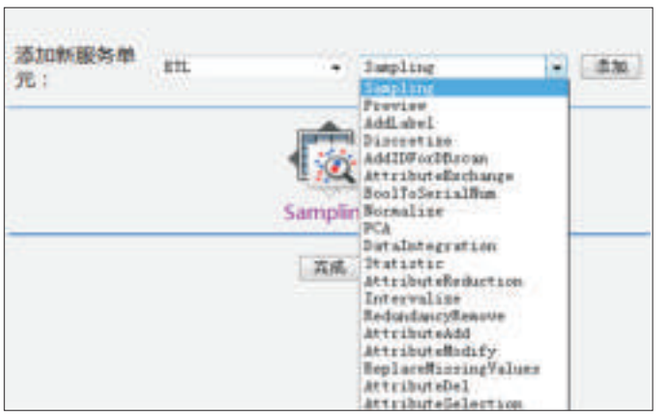


图3▶ 并行 ELT 子系统中包含的各类 ETL 算法

算法、并行分类算法[12]以及并行聚类算法等。

目前该并行数据挖掘子系统中已经开发了很多经典的数据挖掘算法，各类并行算法模块包含的算法如图 4、图 5、图 6 所示，其中并行关联规则算法包括并行 Apriori 算法[13]，并行 FP 树 FPgrowth 以及并行 Awfits 算法；并行分类算法包括并行超曲面分类算法 HSC、并行 k 近邻算法 Knn、并行朴素贝叶斯算法 NaiveBayes，并行决策树算法 C4.5、并行基于范例推理算法 CBR、并行基于类中心算法 CBC 以及并行极限向量机 ESVM 等；并行聚类算法包括并行 DBScan 算法，并行 Clara 算法[14]、并行 k 均值算法 Kmeans[15-16]以及并行 EM 算法等。

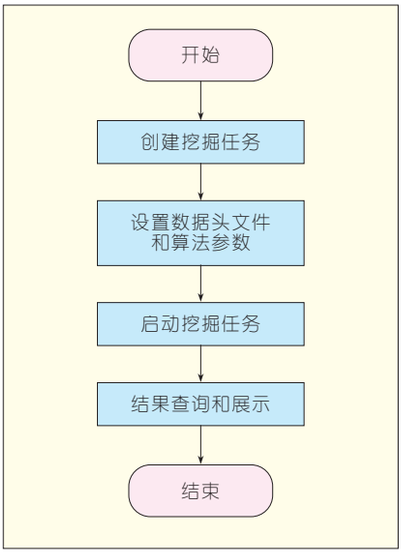
执行数据挖掘算法的一般流程如图 7 所示。从算法流程来看，PDMiner 是一个用户友好的系统，用户不用了解底层算法的设计和实现，就可以很容易使用系统。另外对于并行 ETL 子系统和并行数据挖掘子系统，还提供灵活的接口方便用户集成新的算法。

2.5 基于 MapReduce 实现的算法实例

下面以决策树为例描述基于 MapReduce 的并行算法的实现过程。决策树算法是利用已标记训练集建立决策树模型，然后利用生成的决策树对输入测试数据进行分类。在以前的很多工作，主要是把数据划分到多个计算节点上，然后各自建立决策树模型，最后采用集成的方式得到最终模型[17]。采用 MapReduce 机制可以很好地解决决策树算法内部的并行问题，提高算法的执行效率以及处理数据的规模。

图 8 给出了并行决策树算法的流程图。在该并行算法中，实现了同一层内节点之间、节点内的并行计算，提高算法的执行效率。更重要的是，实现的并行决策树算法以循环代替了递归，使得运行完程序所需要的最大作业（Job）个数可预测（最大数目为样本集中条件属性的数目），从而有利于控制程序的执行状态。而在递归中，无法预测还有多少节点要运算，这样就无法预测程序何时结束。由于层与层之间的运算是串行的，因此在基于 MapReduce 机制的并行决策树实现中，上一层都会传递前缀信息给下一层节点，这些前缀包括从根节点到当前分支的分裂属性信息等。

从流程图可以看到每一层只需要一个 Job，而不关心有多少个节点。程序需要运行的最大层数由条件属性的个数决定，因此是可控制的。由于在并行的过程中主要是统计频率，因此<key, value>的设计非常重要，设置如下：在训练过程中，训练数据被划分到各个节点中进行运算，Map 函数输入的<key, value>分别设计为样本 ID 和样本本身；输出的<key, value>，key 设计为训练样本对应的类别+条件属性的名字+条件属性的值，value 为 key 出现的次数。Reduce 函数的输入和输出的<key, value>的设计均为 Map 函数输出的<key, value>。

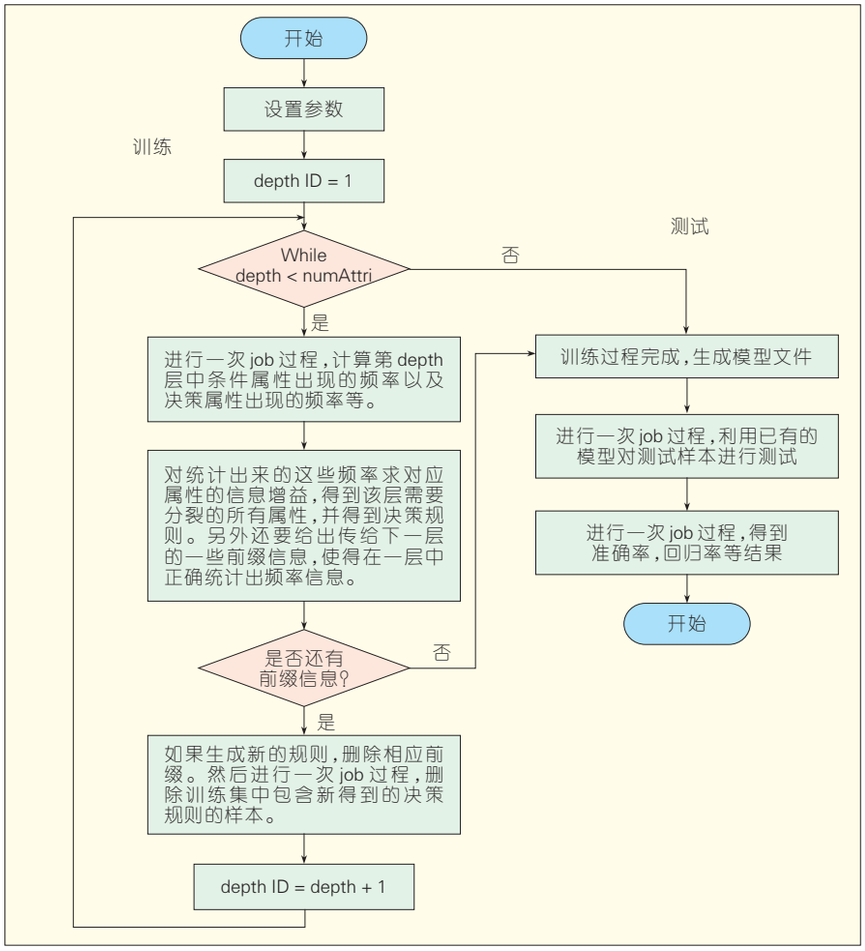


▲图 7 并行数据挖掘算法执行的一般流程

▲图 4 并行数据挖掘算法子系统中包含的并行关联规则算法

◀图 5 并行数据挖掘算法子系统中包含的并行分类算法

◀图 6 并行数据挖掘算法子系统中包含的并行聚类算法



▲图 8 并行决策树算法的流程图（其中 depth 表示树的深度，numAttri 表示属性个数）

当还有前缀的情况下，需要删除训练集中包含生成决策规则的样本，该过程是一个读写的过程。对于包含新得到的决策规则的样本，不再写入训练集，这样在下一次迭代中就只计算那些没有包含生成决策规则的样本。

测试过程则非常简单，每个 Map 利用已生成的决策树模型对样本进行预测，直接样本的预测标记，不需要 Reduce 过程。

3 PDMiner 的特点

3.1 可扩展性

PDMiner 是一个可扩展的并行分布式数据挖掘平台，我们为系统提供了灵活的接口来扩展集成新的并行算法。通过工作流子系统可以很方便地添加一个新的算法，比如在并行 ETL 子系统中添加新的算法 PDAlgorithm1，则只要添加如下代码：

```
#Lists the pdm Filters I want to choose from
pdm.filters.Filter=\
pdm.filters. PDDiscretize,\
pdm.filters. PDNormalize,\
pdm.filters.PDAlgorithm1,\
……
```

通过加入最后一行代码以后就可以在选项卡 PD-Filters 下面加入一项 PDAlgorithm1。生成空类 PDAlgorithm1 的代码如下：

```
public class PDAlgorithm1 extends Filter implements
    OptionHandler, TechnicalInformationHandler,
    Tool, Configurable{
    public PDAlgorithm1( ) { }
    public String[ ] getOptions( ){ }
    public Enumeration listOptions( ){ }
    public void setOptions(String [ ] options) throws
    Exception{ }
  public TechnicalInformation getTechnicalInformation( ){ }
  public int run(String[ ] arg0) throws Exception{ }
  public Configuration getConf( ){ }
  public void setConf(Configuration arg0) { }
  public String getRevision( ){ }
  }
```

其中在函数 listOptions( )、getOptions( )、setOptions( ) 中编写配置算法参数的代码，在 run( ) 函数中编写调用 Map 函数和 Reduce 函数的代码，用户可以根据具体的算法编写相应的 Map 函数和 Reduce 函数。并行数据挖掘算法的添加与 ETL 算法的添加类似。

3.2 支持多挖掘任务

在 PDMiner 中，不仅支持单个任务的创建和执行，而且支持同时创建和运行多个数据挖掘任务。这些任务可以是不同类别的挖掘任务，比如并行关联规则任务、并行分类和聚类任务等，当配置完参数，这些任务可以同时在并行分布式系统 PDMiner 中执行。

支持多挖掘任务功能，具有非常重要的作用。比如要对所有的分类算法进行比较，从而选择对已有数据集表现最佳的算法。一般的做法是串行测试完所有的算法，然后根据算法的效果进行选择。而在 PDMiner 中可以并行地解决该问题，所有的算法都面向同一个数据集（读取同一个头文件信息），最后结果通过系统进行展示，从而选择最合适的算法。从这个比较机制看到，所有的并行算法都是在并行系统中执行，因此可以处理大规模数据；另外，这些算法的执行过程是并行的，评价过程是自动的，因此可以减少算法执行时间和用户的干预。

3.3 创建复杂挖掘过程

通过工作流子系统，系统还支持

创建复杂挖掘任务，可以把并行数据预处理操作和并行数据挖掘算法串联起来。系统提供并行属性删除操作、并行数据归一化以及并行分类算法朴素贝叶斯的串联。当配置完所有算法参数后，其执行过程如下：

• 执行属性删除操作，对数据集进行属性删除操作，并且修改头文件，生成新的头文件信息。

• 接收属性删除后更新后的头文件，进行数据归一化操作。

• 进行分类算法任务。接收从第二步传递过来的头文件信息，然后启动分类算法任务。当任务执行完后，对分类结果进行展示。

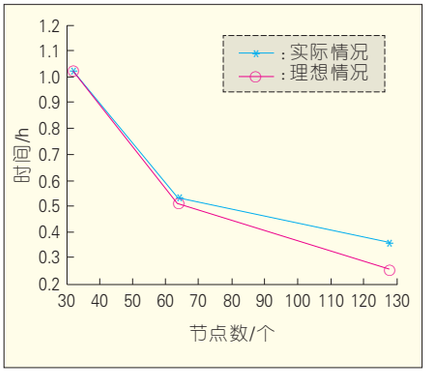
4 实验分析

并行分布式数据挖掘平台PDMiner是一个高效的数据处理与分析工具，主要面向海量数据集的处理。在保证算法正确性的情况下，构造大数据集来考察算法的性能。系统中开发的并行算法已经在通信领域的实际数据挖掘中应用，以下给出了一些算法在构造的大数据集上的性能测试结果。鉴于隐私性等原因，这里没有给出具体的并行算法名称。

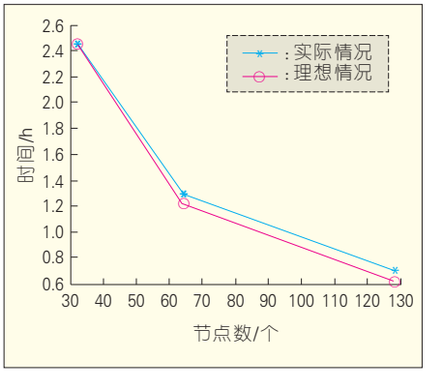
图9、图10、图11、图12、图13给出了2个并行ETL算法和3个并行数据挖掘算法的时间性能。ETL测试的数据规模达到太字节级，而关联规则、分类算法、聚类算法的数据规模分别是30 GB级别、400 GB级别、12 GB级别。我们分别记录了32个节点，64个节点，128个节点的运行时间。若假设32节点执行的时间是标准的理想状态下的时间，图中红线部分给出了理想情况下64节点和128节点的时间性能。从这些图中，可以看到：

• 通过增加节点，都可以提高算法的运算速度，较少执行时间。

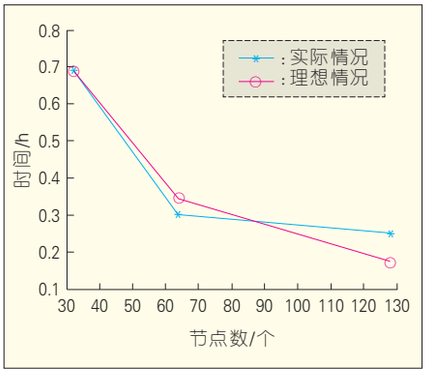
• 算法本身越简单，即并行成分也大，效果越明显，ETL算法显然具有较高的加速比，执行效率也比较高；这说明算法的并行效率与自身可并行化的程度有关。



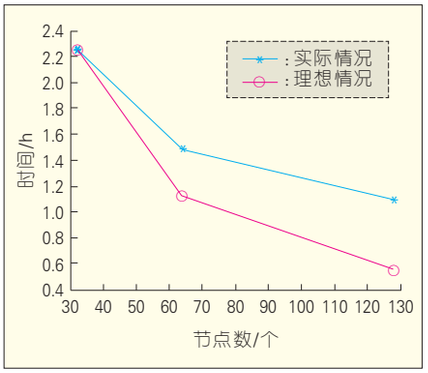
▲图9 并行ETL算法1



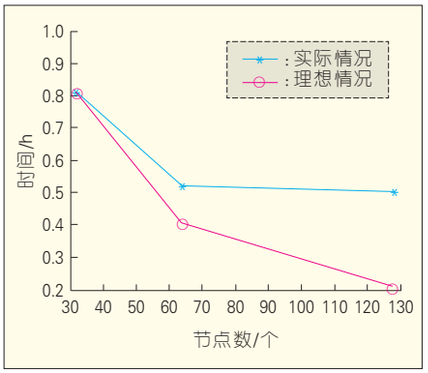
▲图10 并行ETL算法2



▲图11 并行关联规则算法



▲图12 并行分类算法



▲图13 并行聚类算法

• 如图11所示，算法有时候可以得到线性加速比，说明该并行数据挖掘系统可以有效地利用计算资源。但我们也应该看到这种并行计算模型也不是万能的，增加节点并不能总是能很好地提高效果（如图13所示），有时甚至会由于并行通信而使效果变差。

5 结束语

针对大数据的处理和挖掘，本文开发设计了并行分布式数据挖掘平台——PDMiner。基于Hadoop平台和MapReduce的编程模式，开发实现了各种并行数据预处理操作以及并行数据挖掘算法，包括关联规则算法，分类算法以及聚类算法等。另外，PDMiner还开放了灵活的接口，方便集成新的ETL算法和数据挖掘算法。实验测试表明，开发的并行算法可以处理海量数据，且具有很好的加速比性能。

参考文献

[1] HAN J W, KAMBER M, PEI J. Data mining: Concepts and techniques [M]. 3rd ed. San Francisco, CA,USA: Morgan Kaufmann Publishers, 2011.

[2] LUO P, LU K, SHI Z Z, et al. Distributed data mining in grid computing environments [J]. Future Generation Computer Systems, 2007, 23(1):84–91.

[3] LUO P, LU K, HUANG R, et al. A heterogeneous computing system for data mining workflows in multi–agent environments [J]. Expert Systems, 2006,23(5):

258–272.
[4] ZHUANG F Z, HE Q, SHI Z Z. Multi–agent based on automatic evaluation system for classification algorithm [C]//Proceedings of the International Conference on Information and Automation(ICIA'08),Jun 20–23,2008, Zhangjiajie, China. Piscataway, NJ, USA:IEEE, 2008: 264–269.
[5] HAMEENANTTILA T, GUAN X L, CAROTHERS J D, et al. The flexible hypercube: A new fault–tolerant architecture for parallel computing [J]. Journal of Parallel and Distributed Computing, 1996,37(2): 213–220.
[6] GOUDREAU M W, LANG K, RAO S B, et al. Portable and efficient parallel computing using the BSP model [J]. IEEE Transactions on Computers, 1999,48(7):670–689 .
[7] CHU C T, KIM S K, LIN Y A, et al. Map–reduce for machine learning on multicore [C]//Proceedings of the 21st Annual Conference on Neural Information Processing Systems (NIPS'07), Dec 3–6,2007, Vancouver, Canada. Berlin, Germany: Springer–Verlag, 2007:281–288.
[8] BORTHAKUR D. The hadoop distributed file system: Architecture and design [R]. The Apache Software Foundation, 2007.
[9] DEAN J, GHEMAWAT S. MapReduce: Simplified data processing on large clusters [J]. Communications of the ACM, 2008,51(1): 107–113.
[10] 万至臻. 基于MapReduce模型的并行计算平台的设计与实现 [D]. 杭州: 浙江大学, 2008.
[11] HE Q, TAN Q, MA X D, et al. The High–activity parallel implementation of data preprocessing based on MapReduce [C]// Proceedings of the 5th International Conference on Rough Set and Knowledge Technology(RSKT'10), Oct 15–17, 2010, Beijing, China. LNCS 6401. Berlin, Germany: Springer–Verlag, 2010:646–654.
[12] HE Q, ZHUANG F Z, LI J C, et al. Parallel implementation of classification algorithms based on MapReduce [C]//Proceedings of the 5th International Conference on Rough Set and Knowledge Technology(RSKT'10), Oct 15–17, 2010, Beijing, China. LNCS 6401. Berlin, Germany: Springer–Verlag, 2010:655–662.
[13] LI N, ZENG L, HE Q, et al. Parallel implementation of apriori algorithm based on MapReduce [C]//Proceedings of the 13th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel Distributed Computing (SNPD'12), Aug 8–12,2012, Kyoto, Japan. Piscataway, NJ,USA: IEEE, 2012:236–241.
[14] ZHAO W Z, MA H F, HE Q. Parallel K–means clustering based on MapReduce [C]//Proceedings of the1st International Conference on Cloud Computing(CloudCom'09), Dec 1–4, 2009, Beijing, China. LNCS 5931. Berlin, Germany: Springer–Verlag, 2009:674–679.
[15] HE Q, WANG Q, ZHUANG F Z, et al. Parallel CLARANS clustering based on MapReduce [C]//Proceedings of the 3rd International Conference on Machine Learning and Computing (ICMLC'11):Vol 6, Feb 26–28, 2011,Singapore. Piscataway, NJ,USA: IEEE, 2011: 236–240.
[16] HALL M, FRANK E, HOLMES G, et al. The WEKA data mining software: An update [J]. ACM SIGKDD Explorations Newsletter,2009, 11(1):10–18.
[17] 宋晓云, 苏宏升. 一种并行决策树学习算法研究 [J]. 现代电子技术, 2007,30(2): 141–144.

作者简介

何清，中国科学院计算技术研究所研究员、博士生导师；主要研究领域为机器学习、数据挖掘、云计算、并行算法；已承担完成基金项目2项；已发表学术论文近100篇（其中SCI收录27篇，EI收录66篇）。

庄福振，中国科学院计算技术研究所助理研究员；主要研究领域为机器学习、数据挖掘、迁移学习、并行算法；已承担完成基金项目2项；已发表学术论文30余篇。

上接第24页

服务节点，通过cStor系统软件实现统一管理和容错，提供高效、稳定服务。与使用专用服务器相比，可以将系统构建成本节省5～10倍以上，且规模越大，优势越明显。

参考文献

[1] GHEMAWAT S, GOBIOFF H, LEUNG S T. The Google file system [C]//Proceedings of the 19th ACM SIGOPS Symposium on Operating Systems Principles (SOSP'03), Oct 19–22, 2003, Bolton Landing, NY, USA. New York, NY, USA: ACM, 2003:29–43.
[2] SHVACHKO K, KUANG H, RADIA S. The hadoop distributed file system [C]// Proceedings of the IEEE 26th Symposium on Mass Storage Systems and Technologies(MSST'10), May 3–7,2010, Incline Village, NV, USA. Piscataway, NJ, USA. IEEE, 2010: 10p.
[3] SCHWAN P. Lustre: Building a file system for 1000–node clusters [C]//Proceedings of the 2003 Linux Symposium, Jul 23–26, 2003, Ottawa, Ontario. 2003:380–386.
[4] 陈贵海, 李振华. 对等网络: 结构、应用与设计 [M]. 北京: 清华大学出版社, 2007: 83–93.
[5] DABEK F, KAASHOEK M F, KARGER D, et al. Wide–area cooperative storage with CFS [C]// Proceedings of the 18th ACM SIGOPS Symposium on Operating Systems Principles (SOSP'01), Oct 21–24, 2001, Banff, Canada. New York, NY, USA: ACM, 2001: 202–215.
[6] KUBIATOWICZ J, BINDEL D, CHEN Y, et al. OceanStore: Architecture for global–scale persistent storage [C]//Proceedings of the 9th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS'00), Nov13–15, 2000, Cambridge, MA,USA. New York, NY, USA: ACM, 2000:190–201.
[7] RHEA S, EATON P, GEELS D, et al. Pond: The OceanStore prototype [C]//Proceedings of the 2nd USENIX Conference on File and Storage Technologies(FAST'03), Mar 31 – Apr 2, 2003, San Francisco, CA, USA. New York, NY, USA: ACM, 2003:14p.
[8] ZHANG Z, LIAN Q, LIN S, et al. BitVault: A highly reliable distributed data retentionplatform [J]. SIGOPS Operating Systems Review, 2007,41(2):27–36.
[9] BHAGWAN R, TATI K, CHENG Y C, et al. Total recall: System support for automated availability management [C]//Proceedings of the 1st Conference on Symposium on Networked Systems Design and Implementation(NSDI'04), Mar 29–31,2004, San Francisco, CA,USA. Berkeley, CA, USA: USENIX Association, 2004:337–350.
[10] ZHENG W, HU J, LI M. Granary: Architecture of object oriented Internet storage service [C]//Proceedings of the International Conference on on E–Commerce Technology for Dynamic E–Business(CEC EAST'04), Sept13–15 , 2004, Beijing, China. Los Alamitos, CA,USA: IEEE Computer Society, 2004:294–297.
[11] 丁高, 田敬, 陈东, 等. UpStor：一个开放的P2P存储平台 [J]. 计算机研究与发展, 2009, 46(S):250–257.

作者简介

袁高峰，南京师范大学计算机科学与技术专业毕业；南京云创存储科技有限公司云存储项目组项目经理；主要从事云存储产品线的项目管理和研发管理工作。

吴亚洲，河海大学电子科学与技术专业毕业；南京云创存储科技有限公司云存储硬件组项目经理；主要从事超低功耗云存储服务器项目研发管理工作。

薛妍妍，南京航空航天大学通信与信息系统专业硕士毕业；南京云创存储科技有限公司研发工程师；主要从事云存储系统的研发工作。

DOI: 10.3969/j.issn.1009-6868.2013.04.007
网络出版地址：http://www.cnki.net/kcms/detail/34.1228.TN.20130624.1522.004.html

电信大数据解决方案及实践

Telco Big-Data Solution and Experience

中图分类号：TN915.03; TP393.03 文献标志码：A 文章编号：1009-6868 (2013) 04-0039-003

李秋静/LI Qiujing
叶云/YE Yun
（中兴通讯股份有限公司，广东 深圳，518057）
(ZTE Corporation, Shenzhen 518057, China)

摘要： 结合全球多个实际案例，提出了一个电信大数据的精简方案架构。方案结合运营商的实际应用场景，挑选合适的组件进行组合，摒弃了通用化的大平台。大数据的发展，一要通过大数据应用提升运营效率，二要通过数据即服务（DAAS）拓展新的服务内容，提供对外服务。在业务实施设过程中，抓取、管理和挖掘电信运营商的核心数据是基础，运营商大数据的快速部署和应用是最终目标，两者需要在效率、成本和时间上取得平衡。

关键词： 大数据；电信网络；精简架构；数据即服务

Abstract: In this paper, we discuss a number of domestic and international big-data telecommunications architectures and propose our own lean big-data architecture. This new architecture combines the practical application scenarios of operators, and the universal large platform is abandoned. There are two directions in big-data development: improving business efficiency and providing data as a service (DaaS). Capturing, managing, and mining core data of a telecom operator is the basis for service implementation. Rapid deployment and application of big data is the final target. A balance also needs to be struck between in efficiency, cost and time when deploying a big-data architecture.

Key words: big data; telecommunications network; lean architecture; data as a service

1 电信运营商建设大数据思路及关键技术

运营商的网络和用户是运营商的核心资产，而其中流动的数据（包括用户配置基础数据、网络信令数据、网管/日志数据、用户位置数据、终端信息）是运营商的核心数据资产。对于运营商来说，最有价值的数据来自基础电信网络本身，对于基础管道数据的挖掘和分析是运营商大数据挖掘的最重要方向。抓取、管理和挖掘这些数据是运营商的当务之急[1-2]。运营商基于核心数据的大数据应用可从两个方面入手：

（1）通过大数据应用提升自身运营效率。比较典型的应用包括：信令多维分析、网络综合管理及分析、业务和运营支撑系统（BOSS）经营综合分析、精准营销等。

（2）通过数据即服务（DAAS）拓展新的服务内容，提供对外服务。包括个体及群体的位置信息以及用户行为分析等，对于第三方公司（比如零售业或者咨询公司、政府等）都是非常有价值的信息。运营商可以基于这些数据提供对外DAAS服务，拓展市场空间。

为了构建电信运营的大数据应用，从技术能力的角度可以分为数据收集与存储、信息检索汇聚、知识发现以及智慧4个层面。电信大数据技术层面如图1所示。自下而上数据挖掘深度增加，难度加大，对于系统的智能需求提升。其中关键的技术包括抽取转换装载（ETL）、并行计算框架、分布式数据库、分布式文件系统和数据挖掘、机器学习等。

面对海量的大数据，如何有效进行数据处理是需要解决的迫切问题，分布式并行处理是有效手段。传统关系型数据库多采用共享磁盘（Sharing-disk）架构，当数据量达到一定程度，将面临处理的“瓶颈”以及扩展的困难，同时成本也偏高。当前有效的做法是采用分布式文件系统/分布式数据库结合做分布并行处理。目前基于开源的Hadoop平台是业界采用较广泛的一个实现方案。Hadoop[3]的核心思想是基于Hadoop分布式文件系统（HDFS）存储文件或者基于HBase数据库（也是基于HDFS），使用分布式并行计算框架MapReduce来并行执行分发Map操作以及Reduce归约操作。在Hadoop的计算模型中，计算节点与存储节点合一。存储数据的普通PC服务器可以执行MapReduce的任务。而在

收稿日期：2013-04-27
网络出版时间：2013-06-24
基金项目：国家高技术研究发展（“863”）计划（2013AA01A210）

Sharing-disk模型中，存储节点与计算节点是分离的，存储的数据需要传送到计算节点做计算。Hadoop计算模型适合离线批处理的场景，比如Log日志分析、文档统计分析等。它是关系型数据库管理系统（RDBMS）的有益补充。

在私有技术上实现分布式存储和并行处理，在调用接口上与Hadoop兼容，这是一个可行的技术方案。这种方案可以避免上述Hadoop的缺点，同时在性能上做更多的优化。有效的手段包括增加数据本地性（Data Locality）特性，在多次迭代的计算过程减少数据在不同节点之间的传送；使用索引和缓存加快数据的处理速度。结合存储和计算硬件进行调优也是有效的手段，可以使用数据的分层存储，将数据分布在内存、固态硬盘（SSD）、硬盘等不同介质上[4]，使得与计算资源达到很好的平衡。

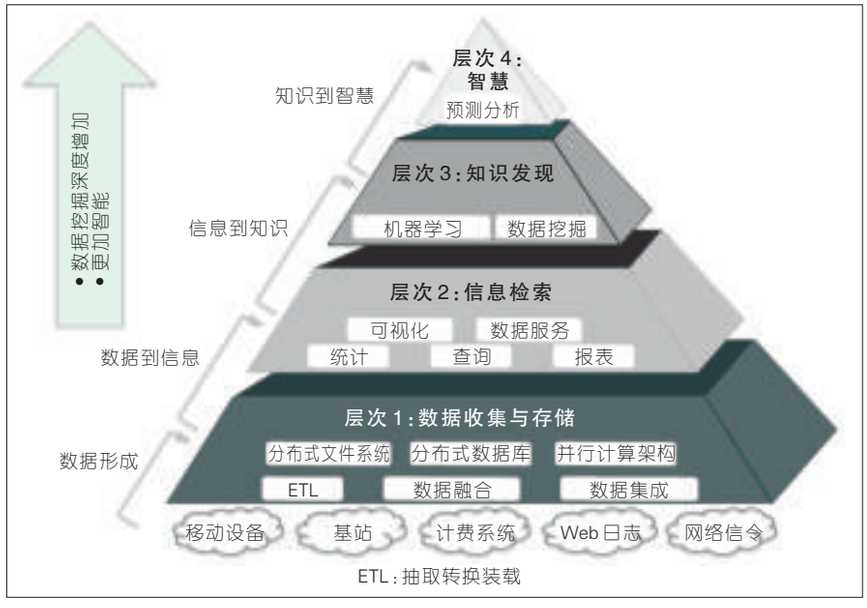
面对海量数据实时性的要求，比较有效的方式是采用复杂事件处理（CEP）[5]。实时流处理采用事件触发机制，对于输入的事件在内存中及时处理。同时对于多个事件能合成一个事件[6]。实时流处理需要支持规则以满足灵活的事件处理要求。实时流处理可以使用分布式内存数据库、消息总线等机制来实现快速实时响应。目前商用的CEP产品有不少，但是在功能、性能以及适用范围上有较大差异，选择成熟度高以及合适的产品是关键。

针对大数据中大量的半结构化或者非结构数据，NoSQL数据库应运而生。NoSQL数据库放弃关系模型，弱化事务，支持海量存储、高可扩展性、高可用及高并发需求。NoSQL数据库在特定应用场景下有很高的优势，是传统数据库的有效补充。按照数据模型，NoSQL主要有四大类：键-值（Key-Value）型、列存储型、文档型、图型，它们对应不同的应用场景。比如Key-Value型适合简单键-值对的高效查询，而图型适合社交关系的存储和高效查询。

针对大数据挖掘分析、搜索以及机器自适应学习等技术在企业系统中逐步应用。相关的算法种类很多，当前需求较多的是分布式挖掘和分布式搜索。

由于数据类型以及数据处理方式的改变，传统ETL已经不适用。运营商需要根据应用场景做不同的规划。目前来说，由于运营商应用系统差别较大，尚未有一种统一的处理模式。比较可行的一种方法是依据数据的功用以及特性做分层处理，比如大量的数据源首先做初筛，初筛完之后有部分数据进入数据仓库或者RDBMS或者其他应用。初筛可以使用Hadoop或者CEP或者定制的方式来完成。

针对运营商的不同应用场景，需要采用不同的技术或者技术组合。比如用户实时详单查询，数据量巨大，但是它的数据类型简单，数据以读为主，不需要复杂的Join操作，数据的分布性好。相比传统的RDBMS，使用Hadoop可以大大提升查询性能，降低处理成本。更多的应用可能需要多种技术的组合。比如信令采集及多维分析，信令数据特别是分组域（PS）信令数据量大且实时性要求高，有效解决海量数据处理与实时性要求是它的关键，需要CEP与Hadoop的组合。在当前阶段，不同的技术成熟度不一，由于业界大数据应用进展较快，我们认为当前针对不同应用的精简方案是最合适的，也就是依据应用场景，挑选最合适的组件做组合，摒弃通用化的大平台。



▲图1 电信大数据技术层面

2 中兴通讯大数据实践

中兴通讯依托在云计算等领域的长期积累，针对大数据形成了一套完整的技术体系架构。ZTE大数据技术体系架构如图2所示。架构依据运营商的不同的应用需求，注重采用组件搭建的方式，形成端到端的精简方案。下面以两个具体的案例进行说明。

（1）用户实时位置信息服务系统

该系统实时采集蜂窝网络用户的动态位置信息，并通过规范接口提供DAAS服务。实际工程中，当期接入的用户数达两千多万，每天用户位置更新数据可达40多亿条，高峰期更新达到每秒几十万次。除了采集的位置，还可以结合其他数据源比如用户年龄等属性做分析，以应用编程

接口(API)开放给上层应用。此外该系统需要有良好的可扩展性,后续可以接入其他区域的数据源。另外这套系统需要有良好的性价比,成本可控,时间可控。依据这些需求,我们在成熟的组件K-V NoSQL数据库的基础上搭建了系统。用户实时位置信息服务系统如图3所示。

用户实时位置信息服务系统是一个典型的精简方案,它基于分布式Key-Value NoSQL数据库的分布式缓存(DCache),组装了对位置流事件实时处理的系统。DCache既是消息总线,也是内存数据库,能很好地满足实时性的要求。同时DCache基于x86刀片服务器,采用分布式架构,系统的扩展性很好,成本较低。该系统性能优越,稳定可靠,取得良好的效果。

(2)信令监测多维分析系统

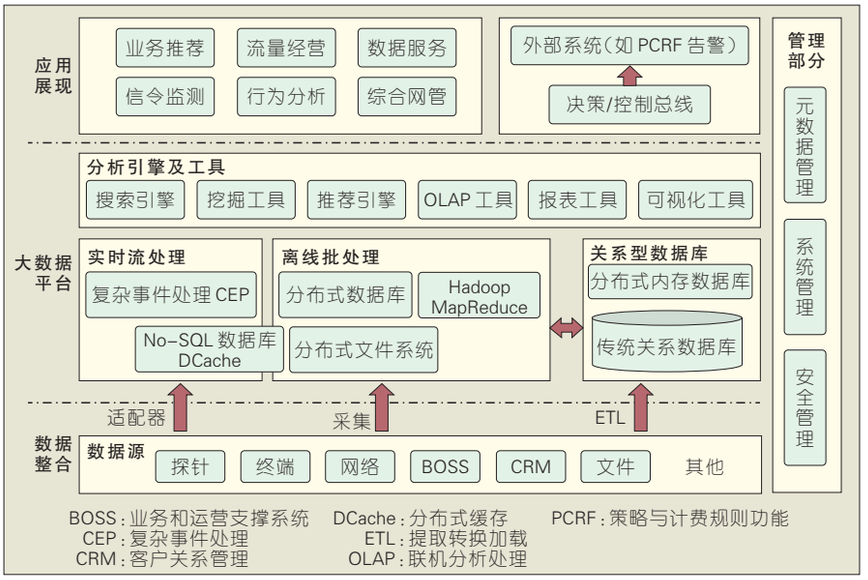
随着运营商数据业务快速增长,运营商对于网络质量提升、网络运营效率有着更大的压力。通过采集网络Gn接口、Mc接口信令并加以处理分析,可以获得网络运行的完整视图,基于信令的相关专题分析,比如网络质量分析、流量效率分析、多网协同分析、客户投诉及服务分析等对于运营商网络运营有极大的价值。

信令监测多维分析的难点在于信令流量大且数据量大,比如某运营商省公司Gn接口峰值流量可以达到4 Gb/s,每天信令数据可达1 TB。需要采集信令并做多种分析以服务于不同的部门。

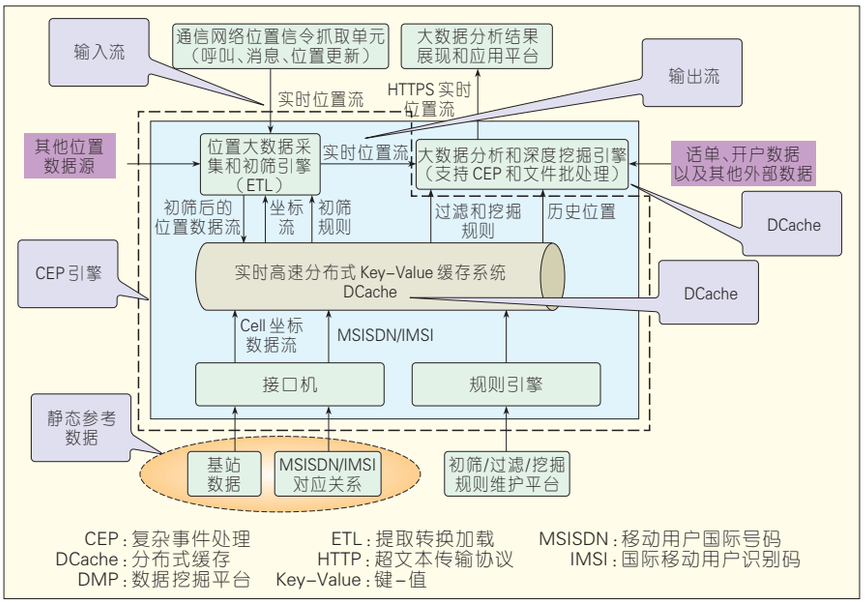
信令监测多维分析系统采用分层的架构,便于数据共享及和应用的扩展。信令监测多维分析系统如图4所示。使用实时流处理满足实时性高的数据分析要求,对于会话或事务详单(XDR)初步处理完的数据采用传统RDBMS存储供后续分析查询使用。对于数据量庞大的XDR采用Hadoop HBase存储并查询,原始信令采用分布式文件系统存放在本地。

在这个方案中,数据根据它的使

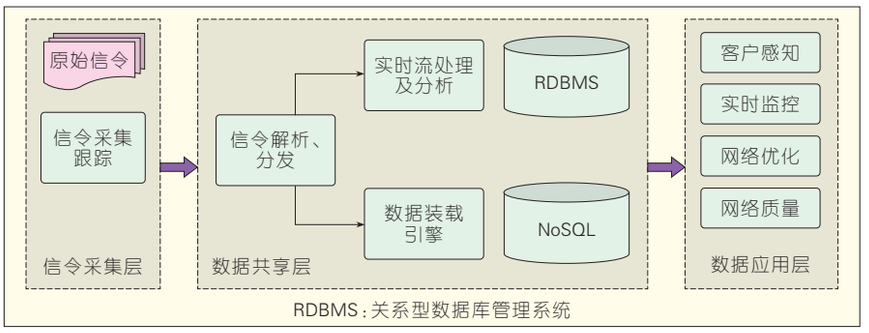
➡下转第45页



▲图2 中兴通讯大数据技术体系架构



▲图3 用户实时位置信息服务系统



▲图4 信令监测多维分析系统

DOI: 10.3969/j.issn.1009-6868.2013.04.008
网络出版地址：http://www.cnki.net/kcms/detail/34.1228.TN.20130624.1516.001.html

面向城市信息感知的社交网络大数据分析

Social Network Big-Data Analysis Based on Urban Information Sensing

中图分类号：TN915.03; TP393.03　文献标志码：A　文章编号：1009-6868 (2013) 04-0042-004

李文俊/LI Wenjun
陆建/LU Jian
王桥/WANG Qiao
（东南大学 信息科学与工程学院，江苏 南京，211186）
(School of Information Science and Engineering, Southeast University, Nanjing 210096, China)

摘要： 基于南京城市信息感知平台，从新浪微博等社交网络数据分析的角度，研究面向城市的信息感知技术；研究结果表明，基于社交网络建立城市规模的计算模型，能迅速感知城市发展的进程，发现城市运行规律，从而实现高效、智能的城市。

关键词： 大数据；城市计算；社交网络；城市感知

Abstract: In this paper, we focus on urban information sensing technology with respect to the Nanjing urban information sensing platform and social network data. Our experimental results show that the urban computing module based on social network data is quite helpful in sensing urban rhythm, discovering the regular pattern, and achieving a more intelligent and efficient city.

Keywords: big data; urban computing; social network; urban sensing

哈佛大学 E.Glaeser 在其新著《Triumph of the city》[1]中指出：城市是人类最伟大的发明，是创新的发动机，城市化让人更加富有、智慧、绿色、健康和幸福。然而，城市化的进程带来了服务与管理上的巨大挑战。如果离开信息技术，城市化很可能演变为巨大的灾难。另一方面，随着移动互联、社交网络、云计算等信息技术的发展，数据在互联网上以远超人们想象的速度迅速膨胀。据统计，全球每秒钟发送 290 万封电子邮件；Twitter 上每天发布 5 000 万条消息；谷歌通过大规模集群及分布式 MapReduce 系统，每天需要处理 24 PB 的数据；淘宝网会员超过 3.7 亿，每天交易量千万笔，产生几十太字节的数据。这些海量数据早已超越了目前人力所能处理的范畴，大数据时代已经来临：企业关注的重点转向了拥有数据的规模以及处理大数据的能力。

近年来的城市计算[2]等技术，受到了极大的关注。在城市计算的概念中，城市空间里的任意设备、车辆、建筑、道路，包括人等都可作为一个计算单元来协同完成一个城市级别的计算。近年来，涌现了一些比较有代表性的工作：在哥本哈根，研究人员通过自行车轮胎上的传感器探知城市空气质量、噪音等[3]；在美国马萨诸塞州，研究人员通过手机用户的通信时刻与位置分析城市动态信息[4]；在北京，微软亚洲研究院的研究者通过分析出租车轨迹研究城市交通问题[5]。

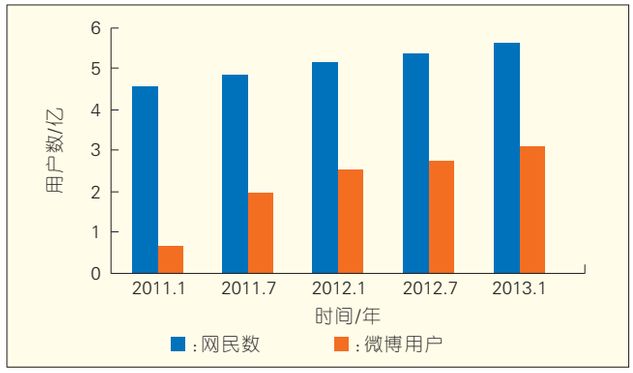
本文依据我们开发的分析平台，通过分析用户在社交网络上产生的数据来感知城市信息。本文旨在展示：依托于网络数据分析尤其是社交网络数据分析，当前已经可以获取城市运行的关键信息，因此可以避免过度把注意力局限到信息采集基础设施的建设方面。

1 社交网络是城市感知的重要途径

据统计，截至 2012 年 12 月底，中国互联网用户达到 5.64 亿，互联网普及率达到了 42.1%。其中，作为新型社交媒体，微博近两年获得了爆炸式的发展，用户规模达到 3.09 亿，较 2010 年底增长了 2.46 亿[6]。图 1 为中国近两年互联网用户及微博用户规模变化示意图。

社交网络的兴起及大量活跃用户的存在，源源不断地产生着大量记录城市生活的数据，这类数据具有交互性、实时性、社会性的特点，隐含着大量有价值的信息，因此社交网络又被誉为“数据科学家眼中的金矿”[7]。社交网络数据的价值引起的许多研

收稿日期：2013-04-23
网络出版时间：2013-06-24
基金项目：国家重点基础研究发展（“973”）规划（2011CB302905）



◀图1
互联网及微博用户规模变化

究者的关注，文献[8]针对社交网络中的大型用户关系网络，提出了一种新的分层社区发现算法；Hao Tu 等人[9]通过聚类方法，对城市热点话题进行检测；Laura Ferrari 等人[10]基于社交网络中的位置信息，通过挖掘频繁模式分析城市信息；文献[11]基于 Google 的 MapReduce 并行框架，通过谱聚类的方法分析社交网络中的用户关系。以上研究从用户关系、言论、位置等方面对社交网络进行了分析，取得了一定的成果，对通过社交网络数据感知城市信息有着非常积极的推动作用。

2 社交网络中的城市信息

本文结合新浪微博数据，自主开发了南京城市信息感知平台，主要完成了以下几个方面的工作。

2.1 城市属性挖掘

在中国 600 多个城市中，既有上海、北京这样的国际大都市，也有丽江、凤凰这样的旅游名城。每个城市都有自己独特的印记和发展轨迹，表现出不同的政治、经济、文化、地理、环境等特征，并反映在城市生活的各个方面。图 2 为江苏省各地级市微博用户活跃度与人均 GDP 比较图。

从图 2 中可以发现，除南京作为政治中心外，其他地级市的微博活跃度与人均 GDP 存在明显的相关性，微博活跃度在一定程度上可以反映出该城市的政治、经济地位。

除微博活跃度外，微博中还包含用户位置、关系、言论等信息，对这些信息进行分析，可以得到更丰富的城市整体及各个区域的政治、经济、文化等属性特征，从而可以帮助人们更好地感知城市、理解城市。

2.2 城市动态性分析

动态性是城市的基本特征，而城市里各个具体对象在位置上的变化，如车辆的运行、人群的移动等，是城市动态性最直接体现。感知城市中移动对象移动的轨迹并对轨迹数据进行分析，可以发现人类社会活动的特征和统计规律，进而可以从微观到宏观的不同尺度上认知和把握纷繁多变的城市动态。

通过对社交网络用户在时间轴上发布的言论、图片等信息进行分析，可以得到用户在空间位置上的变化，比如社交网络中的“签到”功能，支持用户随时记录并分享地理位置信息，提供了丰富的空间移动轨迹数据。图 3 基于社交网络的签到信息对南京不同地点一天中的人流量进行了比较。

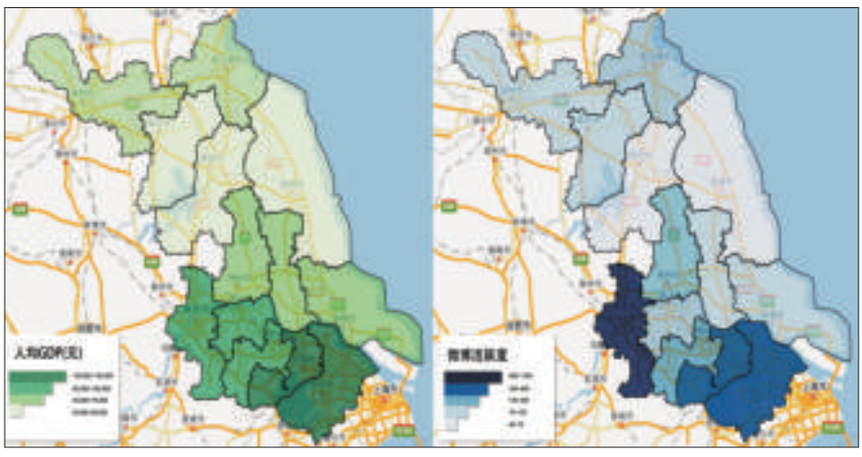
对图 3 进行分析，可以发现景区、餐厅、酒吧的人流量表现出了明显不同的特征。基于位置信息，对城市各空间对象，如道路、商城、小区、医院等动态规律进行分析，有助于人们更好地把握城市动态特征，从而服务于人们的城市生活。

2.3 社区发现

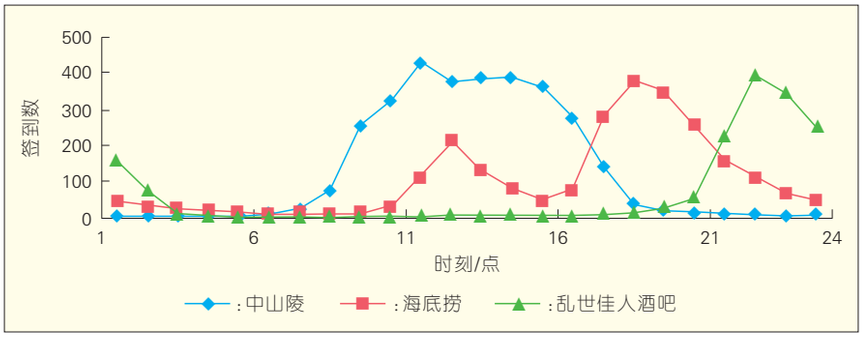
城市是由人组成的，而人类行为大多有潜在的规律。研究表明，人类行为轨迹表现出很强的时间与空间上的相关性[12]，而社交网络中的社区结构同样具有小世界特性，并且表征着人类的共有爱好或者真实世界中的社会关系。

了解人的社交结构，可以通过社交网络中用户间的交互信息，利用谱图技术或者动态社区发现算法[13]完成用户间社区结构的提取，再通过文本分析的技术，分析同一社区的构成原因，如图 4 所示。

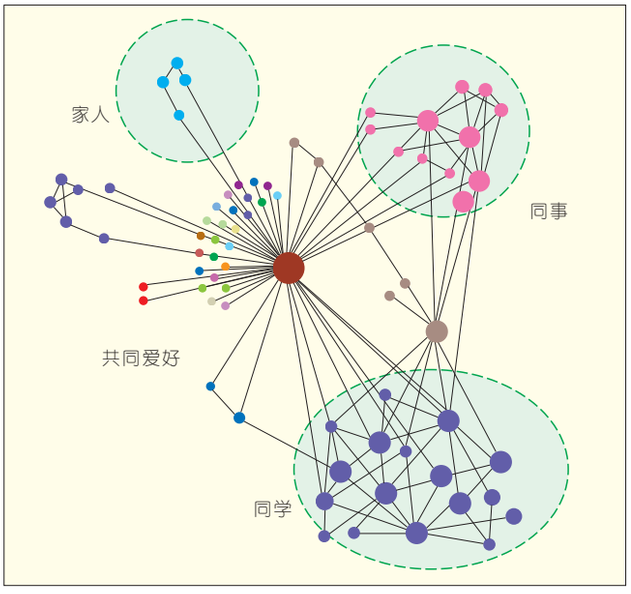
正是由于人类行为的规律性，导致了城市中的种种宏观特征。在数据挖掘更加注重个性化、社交化的今天，从社交网络中挖掘出用户的社交结构和生活模式（行为、意图、经验等），对于研究城市的规律有着极其



▲图2 江苏省各地级市微博活跃度与人均 GDP 比较



▲图3 不同地点24 h签到数比较(景点、餐厅、酒吧)



◀图4
社区发现示意图

重要的意义。

2.4 异常事件检测

异常事件分析是城市计算中的重要研究内容。在城市中,异常事件的发生,如流感爆发、临时交通管制、暴雨灾害等公共事件,往往会对居民生活出行及生命财产等造成损失。

传统的检测手段往往不能及时发现异常事件。以监测流感为例,卫生部门主要通过分析确诊病例来监测流感爆发。由于患者从感染流感到医院确诊通常需要几天时间,这给流感检测带来了时间上的延迟,而社交网络可以为实时监测流感信息提供重要的数据来源。在社交网络中,很多患者在感染流感初期会通过微博发布身体情况,这些信息具有很高的可信度。通过对社交网络中有关流感的数据进行采集、分析,不但可以实时监测流感爆发,还可以预测流感的发展趋势,并及时采取有效的预防和治疗措施。

目前,哈佛医学院的学者[14]通过采集Twitter中的数据来预测流感趋势,并将预测结果与美国疾病预防和控制中心的数据进行比对,获得了比较理想的结果。

除流感外,社交网络在交通事故、群体事件、自然灾害等突发事件的检测中也有着非常重要的作用。社交网络实时性的特点,使其成为检测异常事件的重要手段之一。研究基于社交网络的城市异常事件检测,可以降低异常事件对城市正常运行的影响,减少异常事件给城市居民带来的不便及损失。

3 社交网络数据分析的挑战

社交网络数据是由数亿人在互联网上随机产生的,导致数据杂乱无章,且存在许多重复及无用数据,数据质量偏低。因此,如何从杂乱无章的社交网络数据中,寻找有价值的知识和信息,给科研工作提出了新的挑战和要求:

(1)管理和处理大规模多源异构数据

社交网络数据是典型的多源异构数据,由不同互联网公司产生,且包含图像、文本、声音等多种格式;社交网络数据还包含用户关系、移动轨迹、地理信息、时间序列等各种类型;同时,社交网络包含的数据量非常大,且源源不断地产生大量实时数据,这些都给数据管理和处理带来了很大的挑战。

(2)在线实时分析社交网络数据

许多智慧城市的应用,如城市突发事件检测、城市交通流信息等,有着很高的实时性要求。因此,在对社交网络数据进行分析时,虽然数据量很大,但数据分析过程必须快速高效,以满足实时应用的要求。

(3)如何从杂乱无章的社交网络数据中获取知识

社交网络数据采集成本较低,但同时质量也很低,这要求我们从海量数据中去粗取精,从大数据中提取典型特征;同时单个方面的数据往往只能发现局部的信息量,必须结合多方面的数据去获取更深层次的知识。

(4)如何有效地表达从社交网络中获取的知识并指导人们的决策

社交网络中可以获取城市生活各个角度的信息,但如何合理使用这些信息,将其用于指导城市管理,为人们提供更便利、智能的城市生活,也是比较有挑战的研究课题。

4 结束语

社交网络的兴起为城市感知提

供了丰富的数据来源，但其数据的复杂性也给研究工作带来的诸多挑战。目前的研究工作只是冰山一角，新的研究工作需要转变思维方式，综合各种技术手段，以从纷繁复杂的社交网络数据中发现特定的模式和新的规律，从而帮助人们更好地感知城市信息及发展规律，为人们提供更加美好、绿色、智能的城市生活。

参考文献

[1] GLAESER E L. 城市如何让我们变得更加富有、智慧、绿色、健康和幸福 [M]. 刘润泉, 译. 上海：上海社会科学院出版社, 2012.
[2] PAULOS E, HONICKY R J, HOOKER B. Handbook of research on urban informatics: The practice and promise of the real–time city [M]. Hershey, PA, USA:IGI Global, 2008.
[3] OUTRAM C, RATTI C, BIDERMAN A. The copenhagen wheel: An innovative electric bicycle system that harnesses the power of real–time information and crowd sourcing [C]// Proceedings of the EVER Monaco International Exhibition & Conference on Ecologic Vehicles & Renewable Energies(EVER'10), Mar 25–28,2010, Monaco.
[4] CALABRESE F, PEREIRA F C, DI LORENZO G, et al. The geography of taste: Analyzing cell–phone mobility and social events [C]// Proceedings of the 8th International Conference on Pervasive Computing(Pervasive'10), May 17–20, 2010, Helsinki, Finland. 2010: 22–37.
[5] YUAN J, ZHENG Y, XIE X, et al. Driving with knowledge from the physical world [C]// Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'11), Aug 21–24, 2011, San Diego, CA, USA. New York, NY, USA: ACM, 2011:316–324.
[6] 第31次中国互联网络发展状况统计报告 [R]. 北京: 中国互联网络信息中心, 2013.
[7] BIAN J, AGICHTEIN E, LIU Y, et al. Learning to recognize reliable users and content in social media with coupled mutual reinforcement [C]//Proceedings of the 18th International Conference on World Wide Web (WWW'09), Apr 20–24, 2009, Madrid, Spain. New York, NY, USA:ACM, 2009: 51–60.
[8] LU P, LUO S, HU L, et.al. A novel parallel hierarchical community detection method for large networks [EB/OL]. [2013–02–16]. http://biglearn.org/2012/files/papers/biglearning2012_submission_4.pdf.
[9] TU H, DING J. An efficient clustering algorithm for microblogging hot topic detection [C]//Proceedings of the International Conference on Computer Science & Service System(CSSS'12), Aug 11–13,2012, Nanjing, China. Piscataway, NJ,USA: IEEE, 2012: 738–741.
[10] FERRARI L, ROSI A, MAMEI M, et.al. Extracting urban patterns from location–based social networks [C]// Proceedings of the 3rd ACM SIGSPATIAL International Workshop on Location–based Social Networks(LBSN'11). Nov 1,2011, Chicago, IL,USA. New York, NY, USA: ACM, 2011:9–16.
[11] ZHONG Q, et.al. Parallel spectral clustering based on MapReduce [J]. ZTE Communications, 2013. 2013, 2(11):30–37.
[12] BROCKMANN D, L HUFNAGEL L, GEISEL T. The scaling laws of human travel [J]. Nature, 2006, 439:462–465.
[13] FORTUNATO S. Community detection in graphs [J]. Physics Reports, 2010:75–174.
[14] ACHREKAR H, GANDHE A, LAZARUS R, et al. Predicting Flu trends using Twitter data [C]//Proceedings of the 2011 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS'11), Apr 10–15,2011, Shanghai, China. Piscataway, NJ,USA: IEEE, 2011:702–707.

作者简介

李文俊，东南大学信息科学与工程学院在读博士研究生；研究方向为大数据分析、数据挖掘、Web数据分析等。

陆建，东南大学信息科学与工程学院讲师；研究领域为数据分析和数据压缩；已参与完成基金项目3项；已发表学术论文3篇。

王桥，东南大学教授、博士生导师，东南大学信号与信息处理国家重点学科主任；长期从事信号分析、图像处理以及网络技术研究；已发表学术论文30余篇，出版专著1部。

⬅上接第41页

用特性采用不同的方式存储和处理，突破RDBMS处理“瓶颈”和扩展性的“瓶颈”，达到了很好的效果。在测试中，4节点PC服务器可以全部承担某运营商省公司PS域XDR的存储，入库性能可达50 Mb/s，针对上百亿条记录查询，可以在10 s内返回。取得了很好的实践效果。

3 结束语

电信运营商面临大数据发展的机遇，都在积极推动大数据的试点和商用。在当前大数据技术快速发展的形势下，根据需求和应用场景搭建精简方案，可以帮助运营商在当前激烈竞争环境中快速获得竞争优势，在效率、成本和时间上取得最佳平衡。

参考文献

[1] Cisco Systems. Cisco visual networking index global mobile data traffic forecast update, 2011 – 2016 [EB/OL]. [2013–03–25]. http://www.cisco.com.
[2] MANYIKA J, CHUI M, BROWN B, et al. Big data: The next frontier for innovation, competition, and productivity [R]. McKinsey Global Institute, 2011.
[3] WHITE T. Hadoop权威指南 [M]. 2版. 周敏奇, 王晓玲, 金澈清, 译. 北京：清华大学出版社, 2011.
[4] SNIA. 2012 SNIA Sprint Tutorials–NextGen Infrastructure for Big Data [EB/OL]. [2013–02–15]. http://www.snia.org
[5] NEUMEYER L, ROBBINS B, NAIR A, et al. S4: Distributed stream computing platform [C]// Proceedings of the IEEE International Conference on Data Mining Workshops (ICDMW'10), Dec 14–17,2010, Sydney, Australia .Los Alamitos, CA, USA: IEEE Computer Society, 2010:170–177.
[6] SHARON G, ETZION O. Event–processing network model and implementation [J]. IBM Systems Journal, 2008,47(2):321–334.

作者简介

李秋静，中科院计算所工学博士毕业；现工作于中兴通讯股份有限公司运营商部；主要研究领域为大数据、物联网、云计算等；已发表学术论文8篇。

叶云，中兴通讯股份有限公司运营商部方案总工、高级工程师；长期从事业务软件产品的技术预研、产品方案规划及标准化工作，先后主持和参加了中兴通讯IMS、云计算、物联网、智能管道、大数据等多个重点综合方案的设计；已发表学术论文50余篇。

DOI:10.3969/j.issn.1009-6868.2013.04.009
网络出版地址:http://www.cnki.net/kcms/detail/34.1228.TN.20130625.1727.004.html

对协作系统自适应角色选择策略的思考

Adaptive Role Selection for Cooperative Communication

中图分类号:TN929.5 文献标志码:A 文章编号:1009-6868 (2013) 04-0046-03

摘要:在传统协作系统基础上提出了一种新颖的自适应协作方式——机会角色选择协作。为进一步说明该策略,构建了一种最基本的两用户机会协作框架,其中两个用户相互竞争将各自信息发送给同一目的端。根据瞬时信道条件,两用户均可以机会地充当信息源角色,而另一用户将作为放大转发中继来为信源服务。同时,在此基础上提出并简要描述了一种最优的中心化角色选择策略(C-ROSE)。该策略能够最大化目的端接收信噪比(SNR),进而提高系统传输性能。研究表明,通过这种动态、灵活、高效的节点角色分配方式,协作系统的传输可靠性可得到进一步增强,从而为构建稳固、高效、互惠的协作传输体系开辟了新道路。

关键词:协作通信;中断概率;角色选择

Abstract:This paper describes an adaptive cooperative technique and opportunistic role-selection scheme based on a traditional cooperative scheme. We create an opportunistic cooperative framework where two users compete to transmit their own information to a common destination. Depending on the instantaneous channel conditions, either of the users can be the information source and the other user is an amplify-and-forward relay. To reduce the likelihood of system outages, an optimal centralized role selection scheme called C-ROSE is proposed. This scheme maximizes the received signal-to-noise ratio at the destination. Our dynamic, flexible role-selection scheme can ameliorate transmission reliability in order to create an effective cooperative communication system.

Key words:cooperative communications; outage probability; role selection

葛建华[1]*/GE Jianhua*
丁海洋[2]*/DING Haiyang*
许唐雯[2]*/XU Tangwen*

(1. 西安电子科技大学,陕西 西安 710071;
2. 解放军西安通信学院,陕西 西安 710106)
(1. Xidian University, Xi'an 710071, China;
2. Xi'an Communications Institute, Xi'an 710106, China)

- 机会角色选择协作属于一种自适应协作方式,它是传统协作系统的进化和发展
- 机会角色选择协作可获得满分集增益,能进一步增强协作传输可靠性
- 机会角色选择协作为协作系统研究开辟了新的研究方向

在协作分集系统中,当有多个相同类型的节点(如信源、中继或目的端节点)可供选择时,可通过机会地选取具有最高端到端信噪比的节点参与协作来提高传输鲁棒性。2006年,Bletsas等人针对典型的多中继协作场景首次提出了机会选择思想[1-2]。随后,这种思想被拓展到多源协作系统[3-6]和多目的端协作系统[7-10]中。上述工作的共同点在于:每个节点的角色(即信源、中继或目的端)都是预先确定的,且不随系统中瞬时信道状态的变化而变化。尽管这种预先确定的(固定的)角色配置有其自身优点,它却不能保证每次信息传输时均采用信道质量最好的链路,因此存在一些缺陷:

(1)公平性不足。

(2)能量有效性不足。

针对上述不足之处,文献[11]推导了一种具有两发射机、两接收机的四节点ad-hoc网络的信息理论容量上下界。分析表明:发射机(接收机)间的协作分集可以提供一种高信噪比加性增益。进而,针对相同的系统模型,Ng等人同时考虑了接收机协作和发射机协作,并刻画了其协作开销(以网络中分配的功率和带宽为指标)[12]。最近,通过在上述系统模型

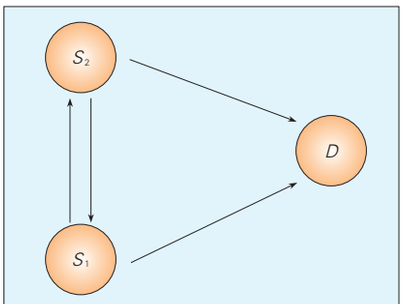
收稿日期:2013-06-02
网络出版时间:2013-06-25
基金项目:国家重点实验室开放课题(ISN12-14)

中增加中继节点并考虑含有多个信源-目的端对的一般场景，Ju等人[13]提出了几种最优和子优的传输策略并分别计算了对应的中断概率。尽管上述工作[11-13]研究了信源或目的端节点间的相互协作，机会角色选择的概念尚未建立起来。更重要的是，机会角色选择的内在工作机理尚未得到深入研究，不同链路对于系统中断性能的影响仍然是未知的。为解决上述问题，文章将构建一种两用户机会角色协作框架，并提出一种中心化的机会角色协作策略。通过这种动态的机会角色选择机制，全面提高系统端到端传输可靠性。

1 系统模型与协议描述

1.1 系统模型

在无线ad-hoc网络中，终端设备间以彼此对等的方式相互通信而不需要有线网络者或基础设施做更好支撑[1]。为克服路径损耗或障碍物等因素影响，信源与目的端间的通信可借助一些中间节点来实现，由此形成了中继链路。同时，由于每个终端设备所具有的能量是有限的，每个节点都会试图借助临近节点帮助转发信息以减少能耗。对于这些场景，机会规划可以有效利用随机信道条件来提高传输鲁棒性，但同时也会带来节点间的相互竞争。如图1所示，考虑一种协作分集系统，其中两个用户S_1和S_2向同一目的端D发送信息。所有终端均为单天线设备且工作于半双工模式。此外，假设任意两节点



▲图1 系统模型

间的信道均满足互逆性且遭受独立的瑞利平坦慢衰落[14]。

在每次两阶段信息传输前，S_1和S_2中的某个用户被机会地选为信息源，另外一个用户作为其放大转发(AF)中继。这里考虑一种能量受限的场景，即S_1和S_2只在两阶段中的某一阶段传输信息。对于该场景，开采S_1和S_2间的协作分集将获得更高的传输可靠性。文章将这种协作机制称为机会角色选择(ROSE)。

1.2 中心化角色选择策略

中心化ROSE策略(C-ROSE)是指在系统中某个节点集中收集所有链路的信道状态信息(CSI)来中心化配置各节点的协作角色。该策略依赖于目的端对CSI的集中式收集和比较判决。具体来说，C-ROSE策略在5个时隙内完成角色选择。在前两个时隙，用户S_1通过直传链路$S_1 \rightarrow D$和两跳中继链路(S_2的放大转发操作将产生另外1 bit信令开销)$S_1 \rightarrow S_2 \rightarrow D$分别传输1 bit测试信令到目的端。进而，目的端执行最大比组合(MRC)来收集直传链路和中继链路信号。类似地，在接下来两个时隙，用户S_2通过直传链路$S_2 \rightarrow D$和中继链路$S_2 \rightarrow S_1 \rightarrow D$分别发送1 bit测试信令给目的端$D$。目的端$D$通过MRC组合收集来自$S_2$的直传链路和中继链路信号，在$D$具有更高组合信噪比的用户被选为信息源，而另外一个用户被选为AF中继。由此，可在目的端D进行中心化角色判决。随后，目的端广播1 bit信令“0”或“1”来告知S_1和S_2其角色判决结果。该过程将占用一个附加时隙，于是产生总共5个时隙的选择延迟。注意，C-ROSE的延迟和信令开销不随系统中瞬时信道条件的变化而变化。

2 机会角色选择的研究意义

机会角色选择机制作为一种崭新的自适应协作方式，它源于传统的固定角色指配机制，但又有其不可比拟的优势：

(1)同样是利用中继协作技术的增益来提高系统的可靠性，机会角色选择机制充分利用了无线信道衰落的随机波动特性对各个节点角色进行实时的最优分配，从而全面提高了端到端的信息传输可靠性。

(2)与传统的固定节点角色机制相比，根据各条链路的CSI，合理地、机会地规划各节点的角色，有助于增强系统中各节点角色配置的动态性和灵活性，进而提高各节点信息传输的公平性。

(3)由于各节点能量或功耗是有限的，通过动态调度各节点参与协作的角色，能够均衡各节点能耗(或功耗)，有效延长系统生命周期，保证系统能量的高效性。

(4)由于ROSE协作系统的节点对等特性和协作自组织特性，可尝试通过分布式ROSE策略以降低机会角色选择的信令开销和实现复杂度。

总之，ROSE协作系统研究是传统协作分集系统研究的深化和拓展，具有重要的理论研究意义和实用价值[12-13]。

3 亟待解决的问题

虽然协作分集系统中的机会角色选择策略有诸多优势，但也存在其特有的、亟待解决的问题：

(1)如何能够降低最佳方案选择的难度？

在机会角色选择协作系统中，每个节点都可能作为信源、中继或目的端，因此候选角色配置方案的种类会随着节点数量的增加而增加。由于最终角色配置方案的确定依赖于各节点间链路的瞬时信道衰落状态，而各角色配置方案共享相同的链路信道状态，这会使得各种角色配置方案之间紧耦合，从而增大候选角色配置方案间的相关性和最佳方案选择的难度。因此，如何在保持节点角色选择动态性和灵活性的同时，最大程度降低方案选择的难度成为一个关键

性问题。

（2）如何通过研究信令反馈、交互传输错误引起的角色判决偏差使理论分析更符合实际应用？

任何理论研究都是以实际应用为最终目的。在角色选择机制协作系统中，大量的信令反馈与交互被引入，为了降低性能分析的复杂度，在分析的过程中往往默认信令传输不会出现差错，或是忽略为支撑信息传输而实际存在的信令传输，由此得到的性能分析结果只能看作是系统实际性能的上界[1-2, 15]，这样做是不全面不客观的，会导致理论分析与实际应用之间存在性能偏差。因此，深入研究信令传输错误对ROSE协作系统性能的影响有助于全面、客观评估ROSE协作系统的实际可达性能，减小理论分析与实际性能间的偏差。

（3）如何合理分配信息和信令发射功率？

在传统的协作系统中，信令反馈和交互较少发生，因此可以在考虑功率分配问题时只关注用于信息传输的发射功率分配，而忽略用于信令传输的发射功耗。然而，对于信令反馈与交互频繁发生的ROSE协作系统，信令交互与信息传输变得同等重要，在系统（或每个节点）总发射功率一定条件下，增大用于信息传输的发射功率就意味着减小用于信令传输的发射功率，信令交互错误会扰乱节点角色的机会规划配置，导致系统传输中断。反之，一味增大信令发射功率虽然能选出最有利于信息传输的角色配置（及对应的无线链路），却会因为信息发射功率的枯竭而对系统传输可靠性产生严重影响。因此合理分配各节点用于信息和信令传输的发射功率，将同时保证信息和信令传输可靠性，最终全面提升ROSE协作系统传输性能。

4 结束语

文章首先构建了一种两用户机会协作框架，在该框架中根据瞬时信道条件，每个用户都能作为信源（或中继）来传输（或转发）信息给目的端。针对该协作框架，我们提出了一种中心化机会角色选择策略，即C-ROSE。可以看出，协作分集系统中机会角色选择策略研究作为一个崭新的前沿课题，可充分研究并利用无线信道的随机波动特性有效提升协作分集系统的抗衰落性能，是传统机会协作机制的拓展和深化。这不仅符合传统协作分集技术的发展趋势，具有重要的理论研究意义，而且对于构建稳固、高效、互惠的协作传输体系也具有重要的实际应用前景。

参考文献

[1] BLETSAS A, KHISTI A, REED D P,et al. A Simple Cooperative Diversity Method Based on Network Path Selection[J]. IEEE Journal on Selected Areas in Communications, 2006, 24 (3): 659–672.
[2] BLETSAS A, SHIN H, WIN M Z. Cooperative Communications with Outage–Optimal Opportunistic Relaying[J]. IEEE Transactions on Wireless Communications,2007, 6(9): 3450–3460.
[3] ZHANG X, WANG W, JI X. Multiuser Diversity in Multiuser Two–Hop Cooperative Relay wireless Networks: System Model and Performance Analysis[J]. IEEE Transactions on Vehicular Technology,2009, 58(2): 1031–1036.
[4] CHEN S, WANG W, ZHANG X. Performance Analysis of Multiuser Diversity in Cooperative Multi–Relay Networks Under Rayleigh–Fading Channels[J].IEEE Transactions on Wireless Communications, 2009, 8(7): 3415–3419.
[5] SUN L, ZHANG T, LU L,et al. On the combination of cooperative diversity and multiuser diversity in multi–source multi–relay wireless networks[J]. IEEE Signal Processing Letters, 2010, 17(6): 535–538.
[6] DING H, GE J, DA COSTA D B,et al. A New Efficient Low–Complexity Scheme for Multi–Source Multi–Relay Cooperative Networks[J]. IEEE Transactions on Vehicular Technology, 2011, 60(2): 716–722.
[7] YANG N, ELKASHLAN M, YUAN J. Impact of Opportunistic Scheduling on Cooperative Dual–Hop Relay Networks[J]. IEEE Transactions on Communications, 2011, 59 (3): 689–694.
[8] YANG N, ELKASHLAN M, YUAN J. Outage Probability of Multiuser Relay Networks in Nakagami–m Fading Channels[J]. IEEE Transactions on Vehicular Technology, 2010, 59(5): 2120–2132.
[9] DING H, GE J, DA COSTA D B, et al. Spectrally Efficient Diversity Exploitation Schemes for Downlink Cooperative Cellular Networks.[J] IEEE Transactions on Vehicular Technology, 2012, 61(1): 386–393.
[10] KIM J, MICHALOPOULOS D S, SCHOBER R. Diversity Analysis of Multiuser Multi–Relay Networks[J]. IEEE Transactions on Wireless Communications, 2011, 10(7): 2380–2389.
[11] HOST–MADSEN A. Capacity Bounds for Cooperative Diversity[J]. IEEE Transactions on Information Theory, 2006, 52(4): 1522–1544.
[12] NG C T K, JINDAL N, GOLDSMITH A J, et al. Capacity Gain From Two–Transmitter and Two–Receiver Cooperation[J]. IEEE Transactions on Information Theory, 2007, 53(10):3822–3827.
[13] JU M, KIM I M, KIM D I. Opportunistic Source/Destination Cooperation in Cooperative Diversity Networks. IEEE Transactions on Wireless Communications, 2010,9(12): 3822–3937.
[14] SKLAR B. Rayleigh Fading Channels in Mobile Digital Communication Systems, Part II: Mitigation[J]. IEEE Communications Magazine,1997, 35(7): 102–109.
[15] DING H, GE J, DA COSTA D B,et al.Link Selection Schemes for Selection Relaying Systems with Transmit Beamforming: New and Efficient Proposals From a Distributed Concept[J]. IEEE Transactions on Vehicular Technology, 2012, 61(2): 533–552.

作者简介

葛建华，现任西安电子科技大学通信工程学院副院长、教授、博士生导师，ISN国家重点实验室副主任，国家数字电视标准化委员会委员；长期从事数字视频广播技术、MIMO-OFDM技术和移动通信技术研究；荣获国家科技进步奖多项；在一些知名学术刊物和IEEE重要国际会议发表学术论文50余篇，完成学术专著多部。

丁海洋，西安电子科技大学博士研究生，解放军西安通信学院讲师；主要从事无线通信技术研究；2013年荣获IEEE通信快报杰出审稿人奖，2012年荣获全军学位与研究生教育研讨会优秀论文一等奖和西安电子科技大学RIM无线研究奖学金；在一些知名学术刊物和IEEE重要国际会议上发表学术论文20余篇，参编德国施普林格出版社学术专著1部。

许唐雯，西安电子科技大学博士研究生；主要从事无线通信技术研究，研究重点为认知协作通信技术；目前已在IEEE通信快报等刊物发表学术论文多篇。

DOI:10.3969/j.issn.1009-6868.2013.03.010
网络出版地址:http://www.cnki.net/kcms/detail/34.1228.TN.20130419.1507.002.html

IPv6网承载NGN和3G业务的测试和研究

Testing on the Capacity of IPv6 to Bear NGN and 3G Services

甘玉玺[1]/GAN Yuxi
金志虎[2]/JIN Zhihu
杨瑾[1]/YANG Jin
(1.中兴通讯股份有限公司,深圳 518057;
2.深圳市芽庄电子有限公司,深圳 518104)
(1.ZTE Corporation, Shenzhen 518057,China;
2. Shenzhen Ya Zhuang Electrion Co., Ltd, Shenzhen 518104, China)

摘要:介绍了中兴通讯与某电信公司联合进行的IPv6和其承载下一代网络(NGN)及3G的测试。详细介绍了3个测试阶段的目的及内容。通过该测试可得知IPv6网络轻载、实施IPv6 QoS和IAD在IPv6网"移动"等,在IPv6网络上承载NGN等业务在技术上实现是可行的。

关键词:IPv6;NGN;3G

Abstract: In this paper, we describe the three stages of tests on IPv6 and its effectiveness in bearing NGN and 3G services. These tests were conducted by ZTE and a well-known telecom company in China. We describe the test objectives and give a detailed account of each stage. An IPv6 network can bear NGN and 3G services and provide a light load, IP QoS assurance, and IAD mobility.

Key words: IPv6; NGN; 3G

中图分类号:TP393.03 文献标志码:A 文章编号:1009-6868 (2013) 04-0049-05

随着互联网规模的持续增长,新需求、新业务的发展[1],2011年2月3日,互联网名称与数字地址分配机构(ICANN)宣布IPv4地址总池已经耗尽。解决该问题唯一有效的办法就是引入IPv6地址体系。

为研究IPv6技术及其承载新型下一代网络(NGN)、3G等业务,中兴通讯于2004年11月开始与华南某著名电信分公司联合进行IPv6及其承载NGN及3G的测试。2004年9月13-17日,广东电信研发中心新技术部也对中兴通讯IPv6网络设备做了测试。

1 IPv6测试的必要性

从IPv4过渡到IPv6,总体的原则是一致的,主要变化是从32位地址变成了128位。从IPv4到IPv6的迁移,将涉及到网络的几乎所有方面,具体包括:终端应用、操作系统、路由器等网络设备、链路层承载以及有关路由和应用协议等。

由于IPv4网络的庞大基础网络、互联网内容提供商(ICP)的规模应用,以及IPv6引入的渐进性,预估从IPv4到IPv6的演进会是一个相当长的过程。由于技术的复杂性以及网络业务的多样性,前期对IPv6的有关情况做摸底和测试是非常必要的。

由于宽带网络业务的发展,IPv4地址的短缺、耗尽以及NGN网络、3G网络的发展需求,具有更大地址空间的IPv6将成为新的核心网络设备所必须支持的功能。其特征包括IPv4、IPv6和多协议标签交换(MPLS)等多协议支持,超大容量的交换、大型协议的支持以及超大路由表容量、高可靠性、安全性以及可扩展性等。

以软交换为代表的NGN,软交换的组网是完全基于数据网络特别是IP分组网络的。从技术需求角度看,软交换对IP承载网主要的要求包括:轻载、高可靠性、高性能、服务质量(QoS)、安全、流量工程等。由于基于软交换的即时通讯等增值ICP业务的大量开展、IPv4地址的缺乏,IPv6将会被使用。而从业务的开展和网络的运维看,数据网络应该支持MPLS及其虚拟专用网络(MPLS VPN)等技术,用于软交换网络和其他网络的隔离。

总之,IPv6的引入对设备供应商和运营商等都是一个巨大的机遇和挑战。

2 IPv6联合测试的目标和阶段

2.1 总体目标

通过双方对IPv6的联合测试,验

收稿日期:2013-02-20
网络出版时间:2013-04-19

证了IPv4/v6双栈设备的有关功能、组网过渡技术和承载NGN、3G等业务，并确立了测试总体目标任务。

- 验证IPv6的有关功能以及其技术先进性；
- 进行过渡组网技术测试：研究将现有的IPv4网络如何迁移到IPv6网络的技术可行性，或者引入IPv6的时机分析等；
- 探讨引入新的业务NGN、3G等在IPv6架构平台的可能性；
- 指导中兴通讯IPv6网络产品进一步优化设计；
- 积累技术经验，为尽早部署IPv6网络做储备；
- 研究新业务和新技术，例如用户可以在华南某著名电信分公司IP城域网内实现移动，不需要改变IP地址等。

2.2 测试3个阶段

IPv6是一种新技术，因此IPv6在新地址体系架构、QoS、移动性、过渡技术和组网以及IPv6本身等方面还处于发展阶段。为了有效地做好IPv6测试工作并平稳过渡到IPv6网络，我们将分3个不同阶段来实施联合测试。

第1阶段：实现IPv6和IPv4互通及现有NGN语音业务承载

- IPv6终端通过双栈网络访问IPv6服务器；
- 双栈路由器ZXR10 T64E采用4in6隧道，并且通过城域网接入IPv6用户；
- 双栈路由器ZXR10 T64E，在条件许可的情况下，通过隧道接入中国下一代互联网（CNGI）或6Bone等IPv6骨干网；
- NAT-PT功能：IPv6和IPv4网络互通，并采用网络地址协议转换NAT-PT；
- IPv6网络和IPv4网络分别接入综合接入设备（IAD），承载语音业务，同时位于IPv4网络SS1能控制接入IPv4网和IPv6网的IAD，提供QoS保证。

第1阶段测试的目的在于：实现IPv6和IPv4网络互通及对原有NGN业务的支持和兼容性，并在IPv6网络承载原有IPv4 NGN业务（通过4in6隧道），另外IPv4 NGN平滑升级IPv6 NGN，同时兼容现网IPv4 NGN业务（IPv6 NGN将在第3阶段考虑测试）。

第2阶段：实现IPv6的相关移动性和QoS

（1）IPv6移动性

- 为常规网络节点和移动网络节点的网络应用提供透明的数据传输（两者都感觉不到移动性的存在）；
- 每一个实现新协议的节点都支持一定的移动功能，以实现整个互联网对移动性的支持；
- 在引入移动性的同时，不能带来安全性方面的问题。

（2）IPv6 QOS

- 测试IPv6基本规范，支持互联网控制消息协议ICMP在IPv6协议下的新版本（ICMPv6）、邻居发现（ND），并实现了“plug and play”（主机和路由器之间自动进行的地址前缀的请求和通告）功能；
- 支持QoS，并且实现QoS的多种机制；
- 测试IPv6多域分类器（MF），分类基于Flow Label、源地址、目的地址、源端口、目的端口、协议号以及流量类别等等，其中地址为128位的IPv6地址；
- 测试IPv6差分服务代码点（DSCP）标注在流量类别字段；
- 测试中兴通讯提供的一个专门的OS封装层，可支持多种OS。

第2阶段测试的目的在于：实现高级性能测试，为实验商用做准备。

第3阶段：实验IPv6小区和承载3G、NGN

（1）寻找一个高档小区，提供两种接入

- 高速上网：上网用户的计算机等终端支持IPv6；
- IAD接入：软交换终端采用支持IPv6 IAD。

（2）面向下一代网络和3G

- 承载支持纯IPv6的NGN测试；
- 承载3G测试。

第3阶段测试的目的在于：进入实验商用阶段。

3 IPv6联合测试的组网方案

IPv6的联合测试组网如图1所示。双栈核心路由器ZXR10 T64E和ZXR10 T128采用单模千兆以太网跨越IPv4的城域网互联，另外采用155M POS3直接互联，以便与ZXR10 G3608高性能双栈路由器形成一个独立的IPv6网络。

两台ZXR10 G3608高性能双栈路由器采用POS3分别接入双栈核心路由器ZXR10 T64E和ZXR10 T128。二层以太网交换机则ZXR10 2826E采用GE分别接入双栈核心路由器ZXR10 T64E和ZXR10 T128。

3G试验网组网如图2所示。位于枢纽机房和科技园机房的两台IPv4/v6双栈核心路由器ZXR10 GER，采用多模POS 155M接口接入IPv4城域网，并采用快速以太网（FE）接口接入核心网（CN）。ZXR10 GER与MSTP的EI接口分别连接无线网络控制器（RNC）、NodeB基站B06C-S111、基站B09-S1/1/1的75欧姆通道化/非通道化E1接口，验证了3G业务在IPv6网络上的移动性等。

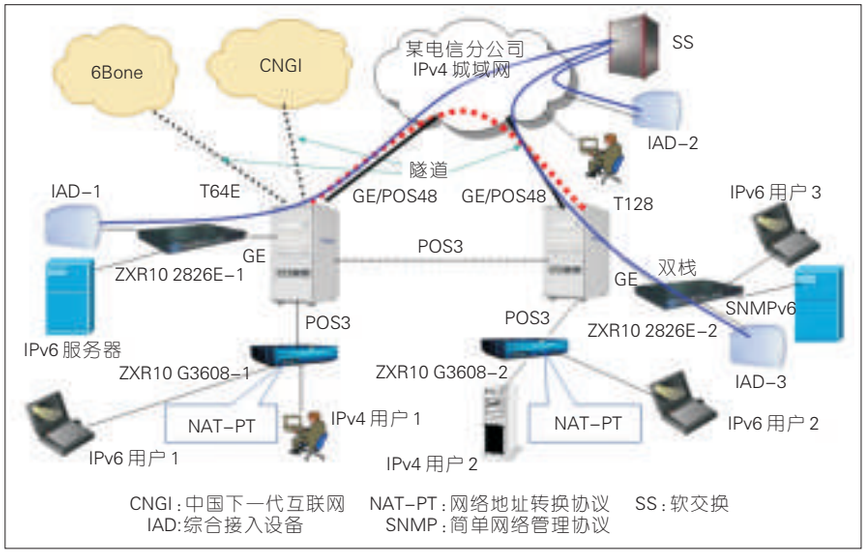
4 第3阶段测试方案介绍

在整个联合测试中，第3阶段涉及到对业务的承载业务测试，是测试的核心目标。

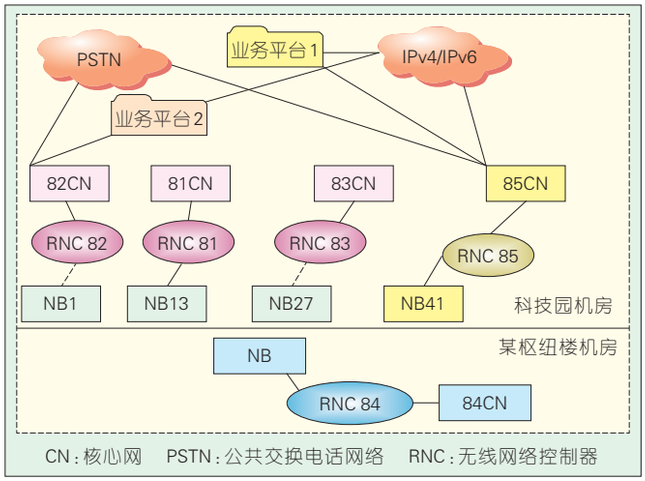
4.1 环境准备

4.1.1 电源动力

- ZXR10 T64E/T128：电源DC、功率：满配置最大功耗<1 600 W；
- ZXR10 GER/G3608：电源DC、功率：满配置最大功耗<400 W；
- ZXR10 2826E：电源DC、功率：



▲图 1 IPv6 联合测试组网



◀图 2 某分公司 3G 试验网组网图

最大功耗<28 W；

- IPv6 的服务器和终端：AC；
- IAD：AC；
- AP：AC。

4.1.2 IPV4 和 IPV6 地址

合作的电信公司为 IAD 和 V4 用户终端提供 20 个 IPv4 地址，并且还可以为设备和 IPv6 用户终端提供 16 个 IPv4 地址。

4.1.3 链路接口

可由 ZXR10 T64E/T128 提供两个 GE 接口：多模 SC 接口 500 m/850 nm、单模 SC 接口 10 km/1 310 nm；或由 T64E/T128 提供两个 POS 48 接口：单模 15 km/1 310 nm、LC 接口，接入现有的 IPv4 城域网。

4.2 测试方案

第 3 阶段测试是对 IPv6 应用的验证和扩展研究，为将来 IPv6 的商用积累了技术和经验。该方案采用在高档小区建立纯 IPv6 网络，并且在这个网络上承载 NGN 相关业务，来实现纯粹的 IPv6 承载 NGN 业务，并和 IPv4 承载的 NGN 实现互通。组网图（如图 3 所示）和第 1 阶段、第 2 阶段的组网类似：只是要在纯 IPv6 网络域内设置一台 SS 控制器。与原先的实现方案不同的是：IPv6 网络域内的 SS 控制器和 IAD 设备要支持 IPv4/v6 双协议栈。

从图 3 可以看出，实现 IPv6 承载 NGN 业务，需有两个重要改变：

（1）IAD 和 SS 控制器须支持 IPv6 协议。IAD 须支持基于 SS 信令的地址配置；SS 须支持基于 SS 信令的地址注册，该信令是支持的 128 位的 IPV6 地址，诸如基于 IPv6 的 SIP 和 H323/248 等信令。

（2）在 IPv6 网内的 IAD 必须能和 IPv4 网络中的基于 IPv4 的 IAD 实现互通。这样就必须在与 SS 相连的 T128 路由器上支持相关的应用层网关技术，实现分别承载于 IPv4 和 IPv6 的 SS 信令之间的转换，同时要配合 NAT-PT 功能为 IPv4 网络域内的 IAD 分配临时的 IPv6 地址等。因此在 T64E/T128 上要实现 NAT-PT + SIP/H323_ALG 功能。

上述是简单的 IPv4 网络域内的 IAD 与 IPv6 域内的 IAD 设备互通所需要的承载层支持，这实际上是 NGN 网络发展的一个起步功能，而更多的 NGN 业务的研究以及 NGN 网络的演进等问题的研究，将在第 3 阶段测试和实验中进行实践研究。所以建议第 3 阶段的具体测试内容可以分成两部分：

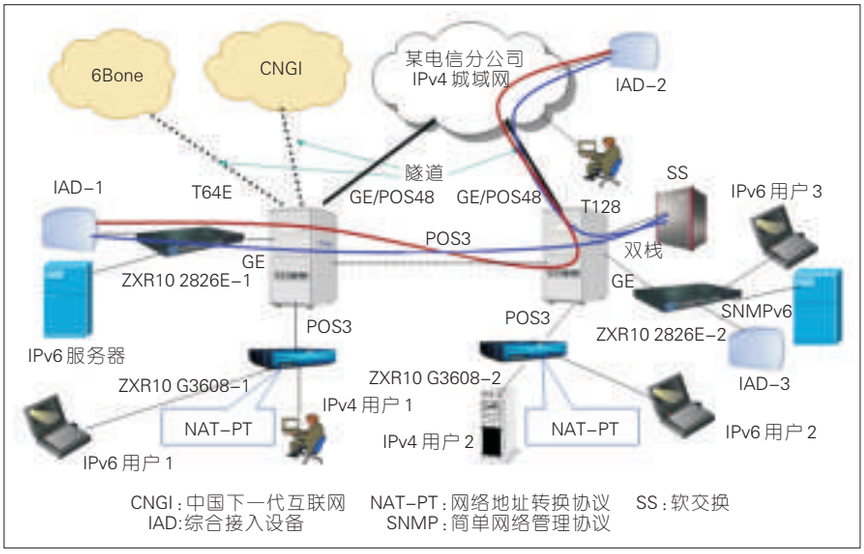
（1）IPv6 承载 NGN 的基本实现方法研究，包括支持 IPv6 的 IAD 和 SS 控制器，IPv6 网络域内的 IAD 设备互通，以及以 IPv6 和 IPv4 网络域内的 IAD 设备互通为目的过渡技术的相关测试。

（2）实现上述功能后，可以将广泛的 NGN 应用和业务在上述网络中进行试验验证，以便对于 NGN 网络的业务开展和网络演进进行实践积累。

第 1 部分内容侧重于 IPv6 网络对 NGN 的承载功能，文章将详细介绍；而第二部分内容则是侧重于 NGN 的网络应用和业务研究。

4.2.1 IPv6 承载 NGN 的基本功能测试

IPv6 承载 NGN 网络在实现上必



▲图3 第3阶段基于IPv6的NGN承载

须满足两点：IAD和SS控制器支持双栈；SIP/H323等信令协议支持IPv6承载。满足这两点基本可以实现IPv6对NGN网络的承载功能。另外，因为存在IPv4承载的NGN网络，因此对过渡技术的支持也是必不可少的，因此诸如SIP/H323的应用层网关技术也必须实现。鉴于此种情况，该部分测试就要侧重于这3点[1]。

（1）IAD和SS控制器支持双栈

首先对于软交换设备包括控制器和IAD支持双栈必须要满足IPv6的基本协议功能，包括：

- 对于IAD设备要支持有状态和无状态地址的自动配置；
- SS控制器支持基于IPv6的地址注册；
- SIP/H323等一些信令支持IPv6的承载；
- 支持号码、网络标示和IPv6地址间的映射和绑定。

（2）IPv6网络域内的IAD设备之间的互通

实验环境如图3所示，测试内容如下：

- 在IPv6网络域内的IAD-1和IAD-3都是双栈设备，SS支持IPv6的地址注册，两者都支持基于IPv6承载的SIP/h323信令，这样就可以实现IAD-1和IAD-3之间的视频电话的相关业务。
- 改变IAD-1的网络接入位置，通过地址的无状态或有状态的自动配置，获得新的IPv6地址，并与原有或新的网络标示或号码进行绑定刷新，使得IAD-1可仍用原来的号码或者使用新的号码与IAD-3实现互通，这样就实现了IAD用户的移动性和ISP的方便快捷的更新。
- 通过对不同的IAD用户的接入端口或者IP地址，设置不同的QOS策略和优先级队列，实现端到端的QOS实现等基本网络功能验证。
- 安全性测试，包括IPSec的实现、用户接入的认证加密实现、访问控制列表（ACL）实现等。

（3）分别位于IPv6和IPv4网络域内的IAD用户的互通

这部分内容主要是检验不同网络域内的IAD用户之间的互通，目的是对未来NGN网络的过渡技术进行实践。因为位于不同网络域内的IAD用户使用的地址格式是不一致的，因此必须要在与SS控制器相连的T128上启动NAT-PT，实现IPv4和IPv6地址的绑定，同时结合SIP/H323等信令协议在不同网络域内的翻译转换网关技术[2]，才能实现两者的互通。测试内容包括：

- 因为涉及到NAT-PT + SIP/H323_ALG功能的实现，必然会对网络的性能产生影响，所以这部分的测试除了功能实现以外，还必须进行性能测试，测试内容包括NAT-PT + SIP/H323_ALG的处理能力测试（包括IPv6地址池的大小、IPv6和IPv4地址映射表的容量、动态映射规则的条目、同时进行会话的条目数等和NAT-PT相关的性能测试）。
- 在启用了NAT-PT + SIP/H323_ALG的端口上的转发性能测试（POS155、GE等接口的线速转发等）。
- 因为NAT-PT的使用破坏了用户的端到端实现，因此对于网络的安全也是这部分实现的重要组成部分，包括ACL、用户认证等。

上述内容只是针对IPv6承载NGN网络的简单实现，而NGN是一个广义的概念，它包含了正在发生的网络构建方式的多种变革。仅作上述的试验还是远远不够的，下面就NGN网络的一些内容，包括它的网络构建特点和新业务的开展进行描述，结合我们的实现网络进行深入测试、探讨和研究。该部分内容包括：

（1）业务能力和业务体系结构

- 阐述NGN提供的电信业务，应用业务与网络分离；
- 建立适当的业务体系结构，着重解决在接口处支持不同的商用模型和不同环境下的无缝通信；
- 考虑后向兼容性，从现有的业务和系统演进。

（2）NGN中服务与网络之间的互操作性

- 复杂系统互操作性实现；
- 标准一致性认证实现。

（3）NGN中寻址功能支持

- 端点寻址，可以使用IP地址和E.164电话号码两种方式；
- 能够将一个E.164号码翻译成端点地址，如一个移动号码的选路基于网络信息或通用个人通信（UPT）号码；

• NGN的IP话机等使用因特网域名系统(DNS)可识别的命名,IP话机选路到特定端点进行呼叫;

• 用户设备能够利用内部数据或外部数据库将用户输入翻译成端点地址;

• 路由协议在NGN功能体系和参考模型中,用户U平面、C平面和M平面流基于IPv6的协议族和机制(例如IPv6地址自动配置、DNS业务、业务的发现等);

• 在NGN范围内考虑IPv6能力的综合有线和无线网络体系;

• 使用IPv6协议及IPv4协议的媒体网关体系和功能模块(包括IPv4与IPv6互通与变换功能);

• 使用基于IPv6控制和管理流的传送网体系。

(4)NGN网络的QOS实验与研究

• 本次IPv6 QoS测试研究提出了基于以太网的IP接入网络的QoS架构,并规定一个分层的参考模型,同时给出协议需求及实施方法。软交换业务的等级采用802.1P COS=1第1类金牌等级[3]。

• IP网络演进成为NGN网络的一种端到端QoS架构,尤其是在核心网络中,MPLS做为一种关键技术而应得以支持,从而方便地提供虚拟专用网络(VPN)、流量工程和服务质量路由。

(5)NGN的一般业务实现

在IPv6网上验证NGN的一般业务,包括3个层面:实现传统的PSTN业务,需要掌握相关的技术,为传统PSTN网络改造和退网设备替换做准备;传输层和承载层提供L1/L2 VPN、电路出租、CDN等业务;固网和移动网的融合业务,要求NGN网络和移动网络在业务层面上能够互通,在网络层面上,能够共享一定的网络资源。

4.2.2 侧重于NGN的网络应用和特色业务测试

NGN特色业务包括:

• 针对集团用户的IP CENTREX、IP VPN、统一通信等;

• 基于ADSL的宽带多媒体的相关业务;

• 针对新建小区的基于LAN的综合业务;

• 针对流动客户群的综合信息超市。

这些内容目前都还是在研究的初级阶段,例如SIP/H323_ALG处于研制阶段,IAD和软交换控制设备SS1b支持IPv6处于开发阶段。这部分实际上是一个针对IPv6承载NGN网络的技术实现和业务开展的长期研究目标,目的是为将来的实验奠定一些基础。双方在该领域建立坚实的合作伙伴关系,共同推进IPv6和NGN网络的实现、发展和演进[4]。

5 结束语

IPv6网络轻载、实施IPv6 QoS和IAD在IPv6网"移动"等,在IPv6网络上承载NGN等业务在技术上实现是可行的。

采用3G业务的IPv6 QoS设为第二等级和双栈路由器承载,部署3G网络在技术上是可行的。

因要穿越IPv4的IP城域网并存在大量的IPv4设备及应用,测试人员需要做大量的配置命令,如第1、2阶段的测试功能项超过54项,配置命令228条,故实现起来比较复杂。

随着4over6软线隧道、NAT444和DS-Lite等新技术不断涌现和10多年的研制及积累,低成本的向IPv6演进已经成熟,并且已列入"十二五"的发展战略,因此向IPv6演进是一个近期的渐进过程。

参考文献

[1] 崔勇, 陈煜驰. 下一代互联网4over6软线隧道技术过渡技术[J]. 中兴通讯技术, 2013,19(2): 21-24

[2] 尚凤军,任宇森,苏畅. 一种快速触发的移动IPv6管理方案[J].重庆邮电学院学报(自然科学版),2010(6),828-833.

[3] 王淑惠,谭清中,唐彦,肖亮. IPv6是物联网最佳的寻址技术[J].数字通信,2011(3),28-31.

[4] 彭龙,胡进峰.基于正斜率增益放大的新型中频放大电器[J]. 雷达科学与技术,2012(3), 112-116.

作者简介

甘玉玺,清华大学硕士毕业;现就职于中兴通讯股份有限公司,任主任工程师,从事推动数据通信、软交换、云计算、LTE-GoTa、IP高清视频会议和绿色动力电源等在政府和企业中商用;已发表论文10篇。

金志虎,日本名古屋产业大学毕业;现就职于深圳市芽庄电子有限公司,任开发工程师,现从事IP数据通信研发和无线产品电路板测试设计等,擅长IT软件开发;已发表论文1篇。

杨瑾,南京邮电大学硕士毕业;现就职于中兴通讯股份有限公司,任主任工程师,从事IP数据通信研究及其知识产权布局等;已发表论文3篇。

综合信息

2013年全球LTE智能手机销量将增至2012年的3倍

北京时间7月8日消息,据韩联社报道,市场研究公司Strategic Analysis本周一发布的数据显示:2013年全球LTE智能手机销量预计将增至去年的3倍。Strategic Analysis表示:2013年将有总计2.7亿部LTE智能手机售出,占据全球智能手机出货总量的29%。 (转载自C114中国通信网)

DOI: 10.3969/j.issn.1009-6868.2013.04.011
网络出版地址：http://www.cnki.net/kcms/detail/34.1228.TN.20130625.1759.005.html

大数据时代的管道技术演进

Evolution of Pipe Technology in the Big-Data Era

朱晓光/ZHU Xiaoguang，陈伟/CHEN Wei，江华/JIANG Hua
（中兴通讯股份有限公司，广东 深圳 518057）
（ZTE Corporation, Shenzhen 518057, China）

中图分类号：TN929.5 文献标志码：A 文章编号：1009-6868 (2013) 04-0054-04

摘要：认为在大数据时代，管道技术将向宽带化演进以提高传输速度；管道架构将向扁平化演进以降低系统延时；软件定义网络的引入使管道向虚拟化发展，实现对网络流量的控制；多管道技术组合则可以使管道向智能化发展。这些方案均可以实现宽带化和实时性传输，实现大数据服务的高速、畅行无阻的传送。

关键词：大数据；管道技术；网络架构；软件定义网络

Abstract: In the big-data era, broadband pipeline technology can improve speed; flattened pipeline architecture can reduce system delay; and software-defined networks (SDN) allow pipelines to be virtually developed so that network traffic can be better controlled; multipipe technology is also used for intelligent pipeline development. All these technologies improve pipe transmission so that big-data services can be transmitted without interruption at high speed.

Key words: big data; pipe technology; network architecture; SDN

随着移动互联网、电子商务、社交媒体、网络视频、企业服务网以及物联网等服务的飞速发展，全球的数据正呈爆炸式的增长。互联网数据中心(IDC)报告指出：2012年已经开始进入大数据时代，2013年全面引爆大数据，2020年全球将拥有共计约35 ZB的海量数据，由此可知我们已经迈进了大数据时代。大数据有4个特征，即超量、高速、多样性和价值，其中高速的特性不仅仅要求大数据能实现实时处理，而且还要求其可以实现实时传送，以增强用户体验。大数据还要依赖于管道网络传输。每个人都是数据的贡献者，同时也是数据的使用者，用户体验要求用户能在任意时间、任意地点接入，并能实现任意呼叫以及在任意浏览地贡献和分享大数据服务，因此大数据将推动管道技术演进，促使管道网络满足大数据的高速畅行无阻传送需求[1]。

数据传送管道满足大数据实时性传送的需求，它主要是通过管道技术演进提升管道传输带宽化，以解决海量大数据时代的数据传输问题。

1 管道技术演进

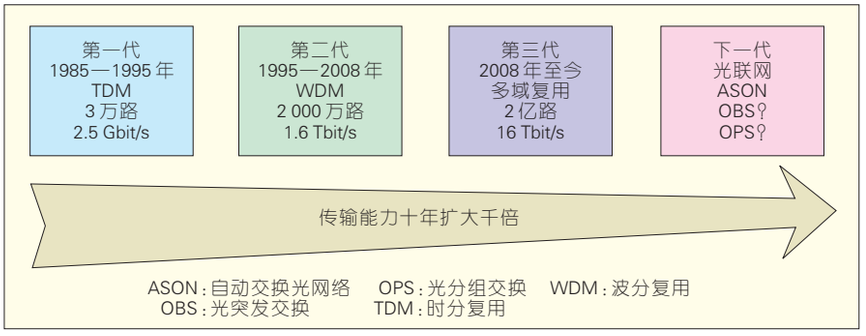
宽带化是管道发展的必然趋势。一方面大数据时代的信息爆炸和海量数据促使传送管道必须越来越宽；另一方面用户体验要求数据的传送必须越来越快。因此无论是光纤传输还是无线接入都要通过技术演进来提高传输效率以实现宽带化。

图1所示为数字光纤传输技术演进过程。第一代数字光纤传输技术采用时分复用(TDM)技术，传输速率达到2.5 Gbit/s；第二代则采用密集波分复用(WDM)技术，传输速率到了1.6 Tbit/s；第三代采用多域复用技术，包括密集型光波复用(DWDM)、正交频分复用(OFDM)、偏振复用(PDM)、正交相移键控(QPSK)和相干检测等技术，传输速率达到了16 Tbit/s；从下一代数字光纤传输到光联网的演进，将采用自动交换光网络技术，传输速率还将大幅提升。因此，在数字光纤传输技术演进中，传输能力每十年增长千倍，同时不断采用新技术，如新的调制技术、相干接收和超强前向纠错等，使光纤的传输能力越来越强。受到大数据时代的强烈需求推动，数字光纤传输将向超高速方向发展[2-3]。

无线传输的宽带化主要通过两方面实现：增加传输频谱带宽以及提高频谱效率。目前无线传输的发展趋势就是空口带宽逐步增宽，如移动通信从3G到长期演进(LTE)再到LTE-A的演进，最明显就是空口占用频谱带宽在增大；在提高频谱效率方面，当前业界主要采用高阶调制和多天线技术，而新技术方面如角动量通信技术还处于初期的研究阶段。另外无线通信的发展受频谱占用等客观因素的影响，技术逐步向高频段发展，如5 GHz、45 GHz、60 GHz、可见光通信等。

移动通信在经过2G基于电路域和窄带技术革新后，就逐步向分组域和移动宽带方向演进，到了LTE阶段，就完全实现基于分组域并且移动宽带化。以移动宽带LTE技术演进为例，如表1所示，其中LTE采用的空口最大频谱带宽是20 MHz，而到了LTE-A阶段，通过载波聚合技术可以

收稿日期：2013-04-14
网络出版时间：2013-06-25



▲图 1 数字光纤传输技术演进

使空口最大频谱带宽达到 100 MHz；在多天线方面，LTE 阶段最高支持 4×4 配置，而 LTE-A 阶段则最高支持 8×8[4-5]。基于上述技术演进，峰值速率从 LTE 阶段的 300 Mbit/s 提高到 LTE-A 的 1 Gbit/s，频谱效率也从 LTE 阶段的 15 bps/Hz 增加到 LTE-A 阶段的 30 bps/Hz。

Wi-Fi 技术的演进如表 2 所示。Wi-Fi 技术从 802.11n 逐渐地演进到了 802.11ac/ad，除了使用频率因客观因素导致的差异外，在技术演进方面，通过增加空口信道频谱传输带宽、采用高阶调制和多天线技术都可以提升 Wi-Fi 的传输能力。Wi-Fi 的传输能力从 802.11n 的 600 Mbit/s 到 802.11ac/ad 的 7 Gbit/s，正向着超宽带传输演进。

2 管道架构演进

在大数据时代，用户体验要求管道网络传输得更快，除了通过技术演进提高传输效率和传输带宽外，还要降低网络延时，减少网元数量和数据交换次数。因此管道网络架构也需要进一步演进，尽可能实现网络简化和扁平化。

图 2 所示为 3 种无源 FTTH 的网络架构和协议，其中图 2(A)是基于时分复用无源光网络(TDM-PON)技术，是当前采用的技术和架构，而图 2(B)和图 2(C)是谷歌光纤网络架构和协议方案，分别是点到点直连到户和基于波分复用无源光网络(WDM-PON)技术。在图 2(A)中，从中心机房到无源光分路器间，多用户共享光纤和带宽，这种架构的缺陷是很难增加带宽，用户各自带宽同时都受限；另外也很难升级网络，因为多用户共享收发器，协议方面采用以太网到 PON 再到以太网的协议栈，需要两次协议栈转换。图 2(B)采用光纤直接从中心机房到用户，即点到点架构，每个用户独占带宽资源，无光分路器，为了降低工程成本，可以通过使用大芯数光缆来实现此方案；图 2(C)则是基于 WDM-PON 技术，采用波分复用技术使每个用户到中心机房都有一根虚拟光纤。图 2(B)和图 2(C)整个网络架构都基于以太网协议，这种架构是一种局域网向城域网络延伸方案，谷歌的这种 P2P 模式是一种跨越式发展，可以使每户带宽达到 1 Gbit/s。

为了降低系统时延，移动通信网络架构也需要向扁平化演进。图 3 所示为移动网络从 3G 到 LTE 的架构变化，从图 3(A)的 3G 网络架构到图 3(B)的 LTE 架构，明显减少了一层网元，因为图 3(B)中 LTE 网络是 eNodeB 直接连接到核心网，而图 3(A)中 3G 网络架构是首先由 NodeB 汇聚到无线网络控制器(RNC)，再进一步汇聚到核心网。这种三层架构不但会造成系统延时长，而且还会降低系统稳定性，增高网络建设和维护成本。

对于互联网架构，可以通过引入内容分发网络(CDN)来提高网络传输性能。尽管互联网架构是基于 IP 协议的网状架构，但是它却受制于管道约束。一方面从信源到信宿除了有物理上的距离外，还要经过多重路由，因此延时不受控制；另一方面大数据传输对骨干网络提出挑战，任何一个环节都可能影响数据传输的速

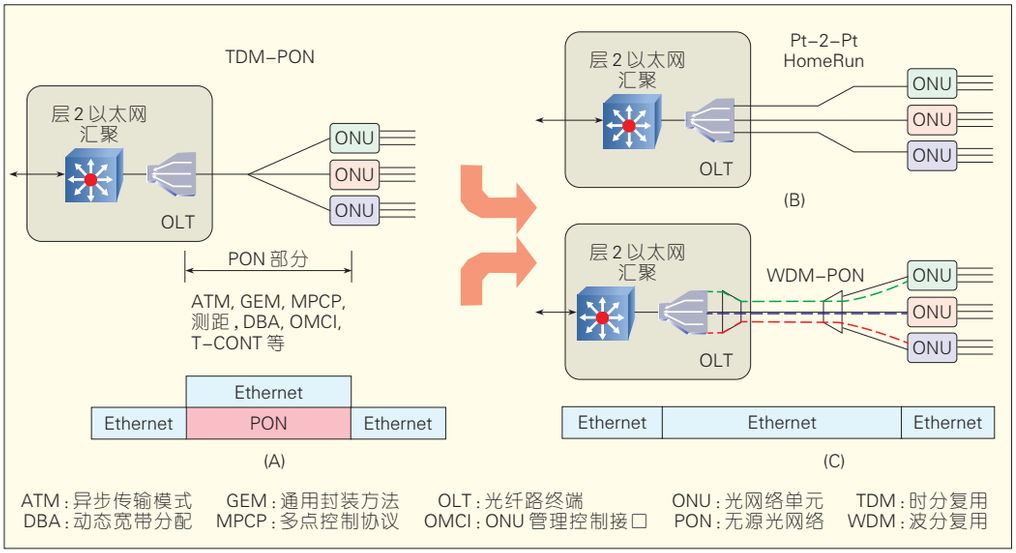
▼表 1 LTE 技术演进对比

	LTE	LTE-A
空口带宽/MHz	1.4、3、5、10、15、20	1.4、3、5、10、15、20、100（通过载波聚合）
多天线	下行最高支持 4×4 配置 上行最高支持 2×2 配置	下行最高支持 8×8 配置，最大 8 层和两码字传输 上行最高支持 4×4 配置，最大 4 层和两码字传输
峰值速率	下行：300 Mbit/s 上行：75 Mbit/s	下行：1 Gbit/s 上行：500 Mbit/s
频谱效率/(bps/Hz)	下行：15 上行：3.75	下行：30 上行：15

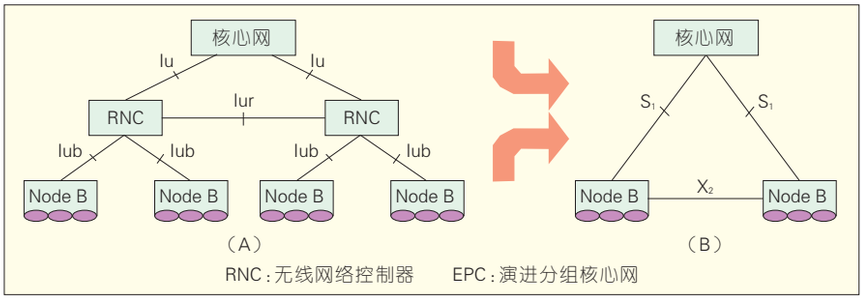
▼表 2 Wi-Fi 技术演进对比

	802.11n	802.11ac	802.11ad
使用频段/GHz	2.4 或 5	5	60
信道带宽	必选：20 MHz 可选：40 MHz	必选：20 MHz、40 MHz、80 MHz 可选：160 MHz、不连续 80+80 MHz	2.16 GHz
调制方式	BPSK、QPSK、16QAM、64QAM	BPSK、QPSK、16QAM、64QAM、256QAM（可选）	旋转调制、差分调制、扩展 QPSK 等改进的调制技术
多天线	必选：1、2 可选：3 或 4，发射波束赋型，时空块编码	必选：1 可选：2 到 8，发射波束赋形、时空编码、多用户 MIMO	自适应波束赋形技术
峰值速率	600 Mbit/s	80 MHz 信道下，峰值达 3.5 Gbit/s，160 MHz 信道下，峰值达 7 Gbit/s	7 Gbit/s

BPSK：双相移相键控 QPSK：正交相移键控 MIMO：多输入多输出



▲图 2 3 种无源 FTTH 的网络架构和协议



▲图 3 移动通信 3G 到 LTE 的网络架构演进

度和稳定性。因此一味提高传输带宽并不能完全解决实际问题。为了使数据传输更快、更稳定，需要在网络中通过增加节点服务器方式，这样使用户就近获取所需内容，解决互联网拥挤问题，提高用户访问网站的相应速度，提升用户体验。

图 4 所示 CDN 网络架构。该架构分为中心节点、区域节点和边缘节点，用户终端就近直接访问边缘节点，不必访问中心节点和区域节点，边缘节点基于缓存服务器，是中心节点的一个透明镜像，距用户仅一跳。因此这种架构，可以降低系统延时，增强用户体验。

CDN 的主要特点是可以使管道带宽得到优化，并且提供自动生成服务器的远程镜像 Cache 服务器，即边缘节点，从而使远程用户可以直接就近访问边缘节点。这样以来一方面可以减少因远程访问带来的带宽需求，分担管道网络流量，减轻中心节点负载；另一方面可以减少网络延时，提高用户存取访问速度，增强用户体验。CDN 的镜像服务，消除不用管道运营商之间互联造成的瓶颈，实现跨运营商的网络加速。广泛分布在管道网络上的 CDN 节点，也是一种节点的冗余备份，可以有效抵抗和降低网络攻击的影响，保证较好的服务质量。

随着大数据时代的来临，CDN 的发展趋势首先是边缘节点逐步下沉，离用户越来越近。在图 5 中可发现 CDN 的发展趋势：CDN 边缘节点距离光纤路终端(OLT)、核心网、网关、边缘路由器等网元设备越来越近。另外未来 CDN 功能集成到一些网元设备中也是一种必然趋势。

总之，管道架构的演进方法无论是通过减少网元数量实现架构扁平化，还是增加 CDN 服务器，都是围绕用户体验来展开的。只有数据以最快速度传送到用户终端，才能让用户享受于大数据时代的高质量服务。

3 软件定义网络

全 IP 网络是管道发展的基础，随着大数据时代的到来，网络越来越庞大，但数据流向也越来越不确定，技术更新和大数据需求要求管道网络有更多的弹性、智能、可扩展性以及自动化能力，因此需要引入全新的网络架构设计理念。在这样的背景下，新网络架构设计引入了软件定义网络 SDN 技术，其核心是将网络设备控制面与数据面分离，网络集中控制，

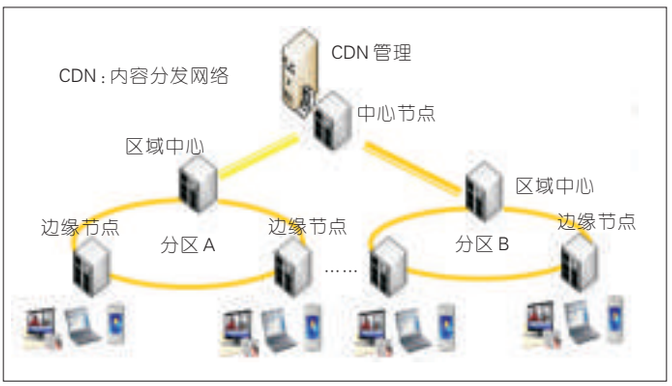


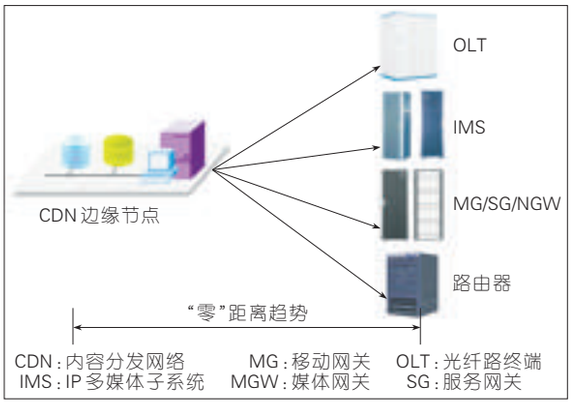
图 4 ▶ CDN 网络架构

资源调度、软硬件解耦以及功能虚拟化等，从而实现对网络流量的灵活调度和智能控制，提升用户体验，提高网络利用效率。

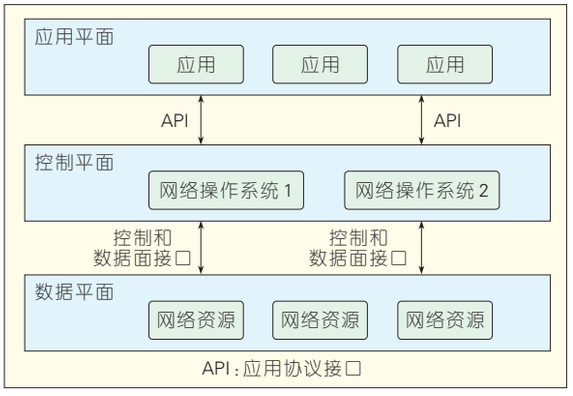
软件定义网络解耦了数据、控制及应用平面，通过支持可编程和分片化来实现转发和控制分离，如图6所示。软件定义网络的主要特征包括：控制转发分离、控制平面集中化、转发平面通用化、软件可编。

软件定义网络实际是将管道虚拟化，使其脱离具体的硬件和厂家设备，并将整个网络变成一个数据转发平台，由控制器统一控制整个网络的资源分配和调度。

软件定义网络使管道设备的软硬件分离，改变了现有管道软硬件捆绑的产业链。软件定义网络一方面降低了对通用硬件的依赖门槛，但也加大了对软件的依赖程度，另外安全问题、集中控制的可靠性问题等都是软件定义网络需要考虑解决的。



▲图5 CDN发展趋势



▲图6 软件定义网络

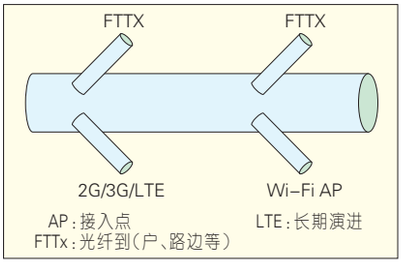
4 多管道组合方式及管道智能化管理

管道技术是多样性的，无论是有线还是无线，单一的管道模式并不能满足大数据时代的需求，多种管道技术以同样的目的但用不同方式向用户提供大数据服务。大数据时代是基于多种管道组合方式向用户传送服务，使用户能在任何时间、任何地点实现任意浏览和任意呼叫。如图7所示，多种细的管道汇聚成更粗的管道，每种管道都将在各自方向演进，提升各自管道的功能和性能，进而使管道变的越来越粗，以满足大数据时代的宽带需求[6]。

管道技术演进一方面满足大数据对带宽化和用户体验的需求，另一方面可以提高管道的利用效率，提升管道的智能化，实现基于用户需求和行为的智能资源匹配。管道运营商也要从粗放型经营转向精细化管理，以支撑新型业务发展，为用户提供按需、灵活的体验和更便捷的个性化服务。管道的智能化还包括多管道间的协同机制，以满足不同用户、在不同应用场景下的不同需求。因此，智能化管理是大数据时代管道技术发展的必然趋势，这样才能合理有效地分配管道资源，提高其利用效率[6]。

5 结束语

大数据时代对其承载的管道技术提出了更高的性能要求，以满足日益增长的用户体验需求，为此，管道技术方面将向宽带化演进以提高传输速度，管道架构方面将扁平化演进以降低系统延时，软件定义网络使管道资源向虚拟化演进，多管道技术组合使管道向智能化管理演进。



▲图7 多管道组合方式

参考文献
[1] LAM C F. FTTH look ahead –– Technologies & architectures[R].Mountain View, CA,USA: Google Inc.
[2] 邬贺铨.大数据时代的网络技术与应用[C]// CCSA第11次会员大会, 2012年12月18日,北京.
[3] 3GPP TS 36.300. Evolved Universal Terrestrial Radio Access (E–UTRA) and Evolved Universal Terrestrial Radio Access Network (E–UTRAN): Overall description[S].
[4] 3GPP TS 25.401: UTRAN Overall Description [S].
[5] 马满仓,郑建勇,郭静,等.WLAN标准IEEE82.11ac/ad及前期关键技术[J].电信技术, 2012(4):75–77.
[6] 赵慧玲,徐向辉. 智能管道发展总体思路探讨[J].中兴通讯技术, 2012(1):4–7.

作者简介

朱晓光，中兴通讯股份有限公司高级工程师；长期从事通信产品研发、技术规划、综合方案、战略规划等工作；累计申请发明专利40余项。

陈伟，中兴通讯股份有限公司一级主任高工、光网络总监；长期从事有线网络技术研究、产品开发和市场规划。

江华，中兴通讯首席架构师、战略规划部副部长、“移动网络和移动多媒体技术国家重点实验室”学术委员会主任；长期从事通信产品的技术规划、应用研究和产业化工作。

DOI:10.3969/j.issn.1009-6868.2013.03.012
网络出版地址:http://www.cnki.net/kcms/detail/34.1228.TN.20130418.1008.001.html

一种分布式复杂消息处理引擎的设计与实现

Design and Implementation of a Distributed Complex Event Processing Engine

中图分类号:TP393.03 文献标志码:A 文章编号:1009-6868 (2013) 04-0058-05

摘要:阐述了一种高性能分布式复杂消息处理引擎的设计方案,这种引擎改进了传统复杂事件处理过程(CEP)处理引擎扩展性问题。新的设计方案通过将分布式无状态数据处理节点与分布式存储相结合,实现了复杂消息处理的规模和性能的线性扩展,同时避免了单点故障,保证了系统的高可靠性。

关键词:复杂事件处理;流式计算;M2M;滑动窗口;实时计算

Abstract: This paper describes a high-performance, distributed, complex event processing engine that improves the scalability of a traditional complex event processing engine. In the design of this new complex event processing (CEP), the stateless processing node is combined with distributed storage so that scale and performance can be linearly expanded. This design prevents single node failure and makes the system highly reliable.

Key words: CEP; stream processing; M2M; sliding window; real-time processing

陆平/LU Ping
钱煜明/QIAN Yuming
朱科支/ZHU Kezhi
(中兴通讯股份有限公司 业务研究院,江苏 南京 210012)
(Communication Service R&D Institute, ZTE Corporation, Nanjing 210012,China)

随着物联网和移动互联网的发展,整个世界已处于数据爆炸的进程中,这也导致了我们认识世界、处理数据的手段不断进步。数年前,各种企业系统还是一个个的信息孤岛,人们研究的重点在于获取信息、打通孤岛,这使得过去十年里人们一直热衷于面向服务的体系结构(SOA)的研究。但在现在这个信息爆炸的时代,每个系统、每个人面临的问题不再是无法获取信息,而是如何能够快速地从海量的信息中获取有价值的内容,并阻止无用的信息淹没有价值的内容。

物联网和互联网应用的一个共同特点是高并发、大数据量,海量消息系统不仅对消息处理的可靠性有一定的要求,对系统扩展性也有较高要求,希望能够从每秒几千次消息到上百万次消息平滑扩展。

电信领域的应用场景采用的实时监测用户信令和行为的方法,例如用户的每一次互联网访问请求、通话、短信、位置变更等信息都需要实时采集处理,并构建用户的行为模型。这个量更加巨大,百万人口的城市信令量就达到每秒数GB的量级,因此靠传统的离线处理基本不可能完成。

目前主要有两种海量实时数据处理方法:第1种方法是通过类似Map-reduce的方法进行在线采集、离线处理;第2种方法是事件流化,直接在内存中进行海量数据的运算和处理。对于消息系统,目前第1种方法有micro-mapreduce[1],它可以将Map-reduce粒度变小,周期缩短,这种方法实时性稍差(5 min-1 h),但能够较好地处理可扩展性问题。第2种方法有现有开源的流式处理框架如S4,商用的产品如Oracle CEP[2]等,该方法能够将相关数据载入内存并进行计算,单机处理性能较高,但处理的可扩展性、容灾容错等存在一些问题,需要在前端进行数据分流,后端进行数据合并。

Storm[3]提供了比较好的分布式解决方案,Storm集群有一个主节点和多个工作节点构成,工作节点与主节点通过Zookeeper协同工作。Storm本质上是一个可靠的分布式消息处理引擎,以保证每条消息都能够被处理。缺点在于其主节点存在单点问题,必须双机HA 2,并且没有时间窗机制,对于事件窗口,以及多路事件协同(例如发生事件A,如果同时过去30 s发生过事件B则生成新的事件C)没有比较好的支持。

对于复杂事件处理(CEP)来说,

收稿日期:2013-02-25
网络出版时间:2013-04-18

提供良好的用户使用界面非常有必要，常用的是使用类结构化查询语言（SQL）的事件处理语言（EPL）来定义事件处理逻辑。Cayuga[4]和Borealis[5]在EPL处理以及事件的服务质量（QoS）处理方面提供了很好的思路。

为达到可靠处理海量实时数据的目的，我们开发了一套全新的高性能分布式复杂消息处理引擎ZX-CEP，重点实现了以下一些能力：

- 复杂事件数据的流式处理；
- 高并发，单机支持每秒十万以上消息量，线性扩展能力较强；
- 简单的EPL消息处理编排以及图形化处理流程编排；
- 分布式计算，系统容量及处理能力的线性扩展；
- 滑动事件窗口。

1 分布式流计算架构

从系统层面看，分布式流计算系统可以认为是一个处理黑盒，大量连续的数据流进入黑盒，经过处理后，转换为特定的事件流输出或传送到其他系统再进行进一步处理。例如系统通过流式处理侦测到某种告警，可以生成告警事件通知自动维护程序进行故障修复操作，也可以将分析后的事件存储到持久化存储引擎以供后续分析处理。

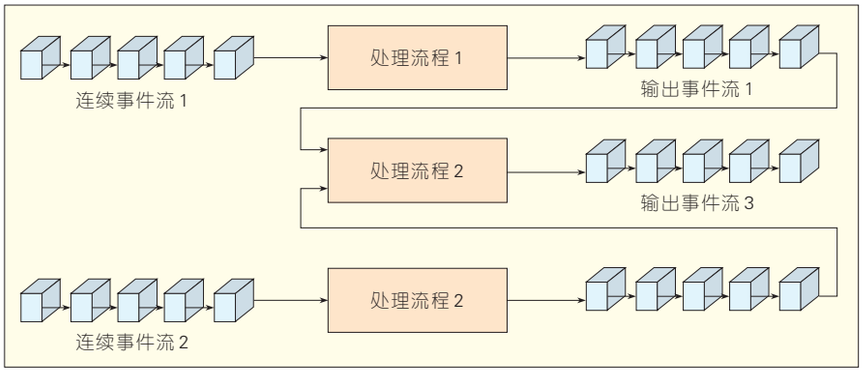
流计算系统内数据的流向本质上是有向无环图（如图1所示），需要对数据进行多重处理的情况下，我们可以将一个流程的输出作为另一个流程的输入，实现多个流程的序列化处理。

分布式复杂消息处理引擎的架构如图2所示。

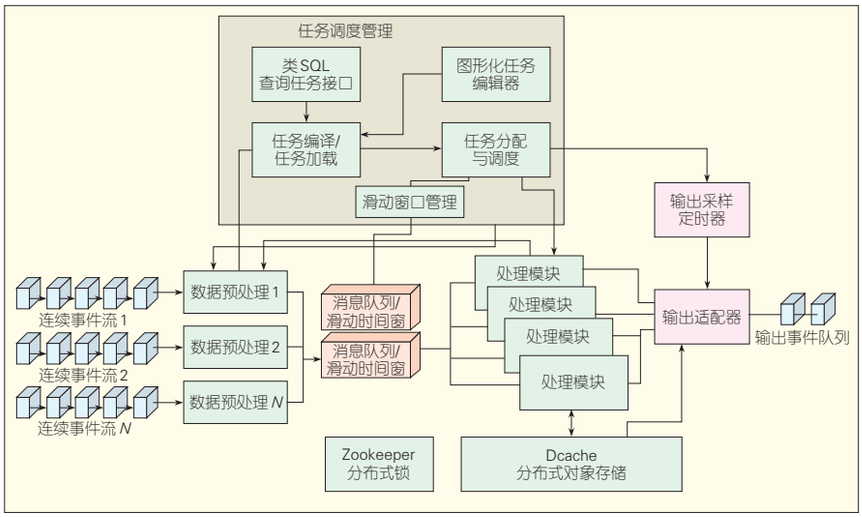
该系统由几个关键网元构成：数据预处理模块、复杂消息处理模块、输出适配模块、任务调度管理模块。

1.1 数据预处理模块

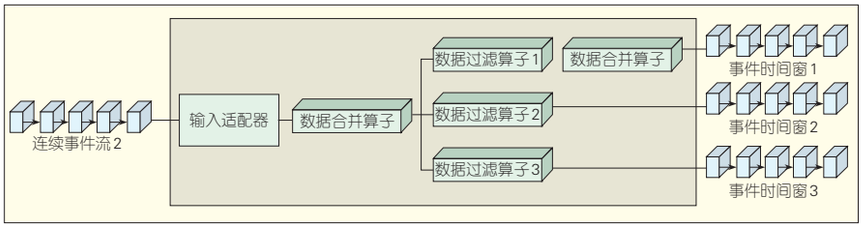
连续事件流传送过来的是各种未经过结构化处理的事件序列，然后再通过事件预处理模块（如图3所示）来实现原始事件的过滤、合并以及分流。



▲图1 多个流程连续处理的数据流图



▲图2 分布式复杂消息处理引擎架构图



▲图3 数据预处理模块结构图

预处理模块又分为两部分：

- 输入数据适配器。该适配器用于接收原始事件序列并转换为结构化事件，并按事件发生的先后顺序送入本地消息队列，等待数据预处理。按照输入内容不同，输入适配器一般需要定制开发。经过输入适配器后，转换为标准的消息体格式，包括消息源ID、消息发生时间戳、消息内容K/V对象。对于无法量化的消息，我们还需要有一个元数据管理，将消息内容进行量化处理映射。
- 预处理操作。我们实现了对事件预处理的一些原子操作，如字段过滤原子、字段填充原子、事件过滤原子、事件合并原子以及事件拆分原子等，通过任务管理器我们实现了基本原子操作的规则定制以及动态加

载。基本原子操作可以实例化为多个算子，各个算子按照定义好的规则进行连接，就可以实现对数据的预处理。各个算子的连接方式，可以通过图形化编辑工具生成，也可以通过EPL语言条件解析产生。对算子操作进行管线化连接的好处是：可以随时对基本算子进行各种串并联操作，实现复杂的数据处理逻辑而不需要复杂代码编写。

在事件处理的过程中，输入信号有可能产生一些超出正常幅度之外的噪音信息，但通过过滤操作我们能够有效去除噪音，保留正常信号[6]。

1.2 复杂消息处理模块

多个事件处理模块侦听同一个或多个窗口变更事件队列，而空闲的事件处理模块则会自动从队列中获取待处理事件。由于事件处理模块本身是无状态的，这样就保证了我们可以随时根据业务情况增加或减少事件处理模块而不会影响到系统的运行。

分布式消息处理的关键有以下两点：

（1）维护分布式消息队列，从而保证事件的序列性。这点我们在DCache K/V系统中已经实现[7]，当然也可以用其他高性能的分布式消息队列实现。如图4所示，通过在分布式存储内维护一致的消息队列，我们可以保证处理的分布式及消息处理的顺序性。

（2）在分布式K/V系统内维护统一的时间窗口。时间窗口由选举出的主节点维护，这避免了各个节点由于时钟不一致而导致的处理误差。

1.3 输出适配模块

输出适配模块用于将系统处理结果转换为特定的输出动作或数据流。输出适配模块有两个基本类型：消息输出以及定期采样输出。当以数据流方式输出时，输出的数据流可以作为输入流并由另一组规则进行后续处理。在这种场景中需要先根据不同纬度的情况进行分析，细粒度观察5 min内数据流情况，并输出整合结果后形成粗粒度数据流，再进行更长时间段范围内的分析（如1 d）输出的方式可以为文件、数据库表或消息队列。输出适配模块一般根据业务需要定制开发。

输出适配模块还有一个功能是时光穿梭，即当规则条件被触发后，通过输出适配模块，我们可以纪录下事件发生前后的系统各种相关消息状况，并做镜像持久化存储，后续可以重放以便分析问题。

1.4 任务调度管理模块

任务调度管理模块的工作流程如图5所示，主要有两部分构成：

（1）规则的生成。我们可以通过两种方式生成规则，一种是通过EPL的事件处理语句，动态定制生成任务图；另外一种则是通过规则编辑器，以图形界面方式生成事件处理逻辑。

（2）规则的调度执行。在业务过程中，我们需要动态的规则加载，也即规则加载过程不能够影响正常的处理过程。EPL适合比较简单的规则场景，规则图编辑则适合比较复杂的规则场景。为了提升效率，我们做了图形化的规则编辑器，将规则图生成后直接转换为对应的代码实现并实现了程序代码的动态加载。

当指定一个滑动窗口将被适配新的规则时，存在如何匹配发生在规则生效前旧数据的问题。在此我们定义了两种实现策略：一种是新规则部署后我们将清空对应窗口数据，但这可能会导致数据有一定时间中断；另外一种策略是我们记录新规则生效后的时间信息，在此期间内新旧两套规则同时计算，当新规则生效后的数据出栈后，我们才正式启用新规则的计算结果，否则一直采用老规则计算结果。这可能造成的影响是仅当$T=Tw+1$（W为时间窗宽度）时间后新的规则才能够生效。

2 滑动窗口设计及脏数据污点传播机制

通过高性能分布式消息队列我们可以实现滑动事件窗。滑动事件窗的定义是一个唯一的事件序列，需要在系统中保留固定的时间长度或者固定数量的消息，随着时间推移，仍保持在该事件序列内的所有消息都维持在特定时间/长度范围内。因此滑动事件窗分为两类，一类是滑动时间窗，所有事件都维持在特定的时间区间内；另一类是滑动空间窗，预先定义好窗体内事件的容量，超出容量后的事件将自动出栈，如图6中所示。

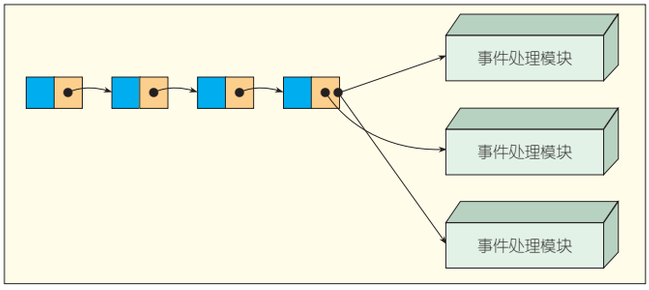


图4 多个事件处理模块分布式处理同一变更队列内容

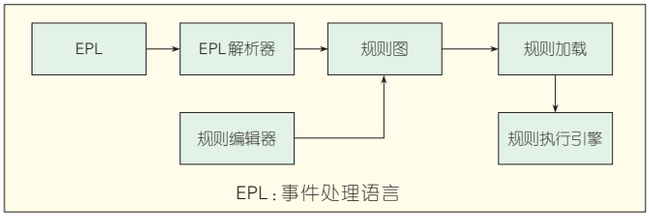
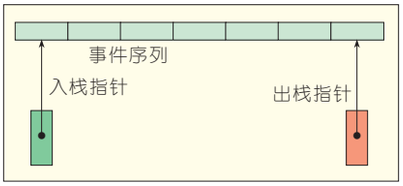


图5 规则的生成与加载



▲图6 分布式滑动事件窗口

为了保证系统的分布式处理，我们采用了分布式K/V引擎来维护滑动事件窗口，这样事件的一致性存储就由分布式K/V引擎来保障。

事件序列的每个元素，以及入栈指针和出栈指针均作为键值对保存在分布式K/V引擎中，这样我们就实现了分布式的滑动事件窗口存储，如图6所示。其中入栈指针和出栈指针我们使用了特定的同步操作模式来进行存取，保证了在分布式环境下的数据一致性。

每个事件进入事件窗，或者定时扫描发现是否有事件退出事件窗口，都会激发消息处理动作。该动作会激发复杂消息处理模块进行处理。为了保证处理的分布式，这里采用了消息队列方式，实现拉模式的消息处理。滑动事件窗的进入、退出事件会生成窗口变更消息，该消息会进入另一个消息队列等待复杂消息处理模块响应处理。

为此我们构建了类似Aurora[9]的数据模型，并将时间窗的事件序列转换成为增量事件序列。3种增量事件分别为：

- 插入事件：(+, t) t为新增到事件窗口的事件对象；
- 删除事件：(–, t) t为从事件窗口退出的事件对象；
- 替换事件：(^, t_1, t_2) t_1为被替换事件对象，t_2为新事件对象。

通过处理增量事件，系统能够有效避免经常性的全局扫描事件窗，从而大大加速处理的进程。在事件信息中，我们还增加QoS标识，并发送到不同优先级的队列中，这样可以保证高优先级事件被优先处理。我们利用分布式K/V存储维护了事件状态机以及全局计数器，在事件处理过程中，有效简化了数据的处理逻辑。以最简单的计算事件窗内所有事件的平均值为例。普通方法是每次事件都需要重新计算时间窗内所有事件的平均值：

$$T(k)=\left(\sum_{k=0}^{n}E(k)\right)/n$$

而通过增量事件后，每次则需要计算：

$$T(k+i)=T(k)+\sum_{m=k}^{k+i}E(m)/i$$

如果采样周期为1 s，事件窗则为5 min，则后一种的计算量就只有前一方式的1/300。

计算的分布式带来一个额外的问题：对于复杂的计算，有可能涉及多个事件序列，因此多个事件队列产生的事件并不一定由同一个事件处理器处理。在此我们引入了计算的污点数据传播模型，以保证任何一个基础事件带来的信息更新都能够及时引发后续处理节点的处理。

当涉及到某一个规则需要使用两个或多个滑动窗口内的数据时，因为我们的系统是分布式处理，就导致了有可能两个滑动窗口产生的事件流并不是在同一个节点上进行的分析处理。为此我们设计了分布式的污点数据传播机制[5]，保证一个规则数的各个处理节点都能够对最终结果进行正确更新，即使并非在同一个节点完成的计算。如图7所示，灰色部分数据代表脏数据，通过数据传播机制来传递脏数据标识，从而保证所有数据及时得到更新计算。

整个规则树可以认为是一个有向无环图，变化后的数据经过算子后，有可能影响后续处理节点，也有可能没有影响。动态分析能够查询出所有被影响节点，静态分析仅能够分析出可信边界，可信边界外部数据可能是被污染，也可能是疑似污染，可信边界内部的数据可以确保是干净的。

应用污点传播算法，我们可以识别出受影响的节点，并应用算子重新计算受影响节点的后续数据，而对于没有受影响的数据则不需要重复计算。作为分布式系统，每个数据以及算子，都有可能发生或保存在不同的物理节点中。

污点传播的分析有静态分析与动态分析两种，本系统实现了编译基本的静态分析。当源数据发生改变时，系统可以分析出后续有向图中所有被影响的节点，并标示出被影响数据。当需要获取被影响数据进行计算时，根据被污染标示，可以进行前向逆推计算。这种情况下保证了整个系统的计算量最小。

污点检查策略的首要任务就是分析可信边界，污点检查策略表示为一个由实体类型(type)、脆弱性描述(vul)、程序操作(op)以及操作数位置(loc)组成的4元组：

$[type, vul, op, loc] \mid type \in ROLES, vul \in VUL\_TYPES, op \in ACTS, loc \in \{N \cup any\}$。

针对每个输入变量，对应每个计算节点进行污染检查，我们就可以整理出系统的污染传播矩阵，如表1中所示。

对于多输入变量环境，被污染节点和疑似污染节点是单输入变量的并集。通过污点传播算法，我们可以

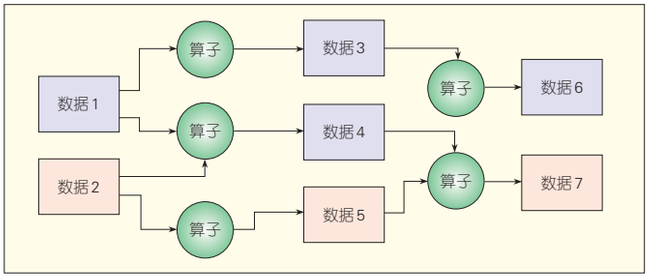


图7▶ 污点传播的动态数据运算

▼表1 系统的污染传播矩阵

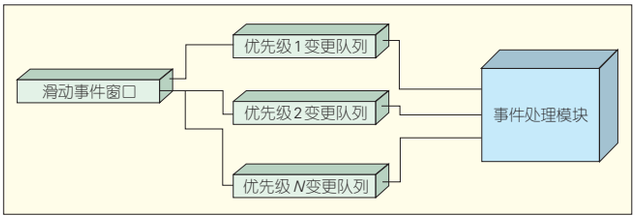
输入变量	被污染节点	疑似污染节点
A	T_1, T_2	V_1
B	T_2	V_1, V_2

让系统只在需要输出数据的时间点对于脏数据节点进行数据更新计算操作，而不需要时刻全面更新系统数据节点，这样能够极大降低系统的计算量。

3 事件处理的QoS保障

增量事件消息队列不止一个，根据QoS标识不同，不同优先级别的消息会被放入不同的增量事件消息队列。通过这种方式，我们能够实现优先处理高优先级事件信息。

事件处理模块优先从高优先级队列获取变更消息（如图8所示），高优先级队列中没有待处理信息后再从低优先级队列获取信息。



◀图8 事件处理的QoS保障

事件处理模块的处理结果可以暂存到DCache中，或输出到输出队列。对于暂存在DCache中的计算结果，另有一个采样工具定期采样并输出到输出队列。例如我们需要统计某一个传感器组过去1小时窗口内平均值，并且每5分钟报告一次，同时一旦发现某时刻读数过高则需要马上发送告警。这时事件处理模块对于每一个新的输入输出事件，都会修改维护在DCache中的平均值对象。采样程序每隔5分钟从DCache中采样该平均值数据，并输出到输出队列，对于读数过高的数据则即时生成告警事件并放入输出队列。

4 结束语

文章描述了ZX-CEP分布式复杂消息处理引擎的设计及实现，该引擎能够高性能实时处理复杂的流式数据。我们首先基于数据与逻辑分离的原则对该系统进行了设计，数据存储节点采用云存储方式，保留多副本；数据处理节点采用无状态节点，可以分布式动态进行扩展。该架构既保证了海量数据下的存储可扩展性以及数据安全性，也保证了并行处理下的计算可扩展性。同时该架构还保证了任意一个节点故障对于系统业务正常处理没有任何影响，流式计算仍然能够持续进行而不会被中断。

本架构依赖于分布式K/V存储以及构建于分布式K/V之上的分布式消息队列，并通过分布式消息队列实现了跨节点共享的滑动时间窗。

我们展现了使用EPL语言实现对于数据处理逻辑的实时定制与加载机制。通过EPL完成基于基础算子之上的复杂逻辑编排图。由于分布式数据处理的特性，数据的分布式处理及存储带来了分布式逻辑运算的复杂性，因此我们引入了脏数据传播机制，让数据驱动处理逻辑。

未来我们将致力于进一步提升本系统的动态逻辑处理机制，让逻辑判断更加灵活，支持更加复杂的逻辑运算。同时我们将提升本系统的可维护性，确保能够自动发现故障，并通过调整数据存储及计算节点实现故障的自我修复。

参考文献

[1] CONDIE T, ALVARO P, HELLERSTEIN J M,et al. MapReduce online[R]. UCB/EECS-2009-136.Berkeley,CA,USA: University of California, Berkeley,2009.
[2] 黄强，增庆凯.基于信息流策略的污点传播分析及动态验证[J]. 软件学报,2011, 22(9): 2036-2048.
[3] 李新玉,黄忠东.基于CEP的可持久化事件处理方案[J].计算机应用与软件, 2010,27(12): 151-153.
[4] CHERNIACK M, BALAKRISHNAN H, BALAZINSKA M, et al.Scalable distributed stream processing[C]//Proceedings of the 1st Biennial Conference on Innovative Data Systems Research (CIDR'03),Jan 5-8,2003, Asilomar, CA, USA. New York,NY,USA: ACM, 2003:12p.
[5] ABADI D J, AHMAD Y, BALAZINSKA M,et al. The Design of the Borealis Stream Processing Engine[C]//Proceedings of the 2nd Biennial Conference on Innovative Data Systems Research (CIDR'05),Jan 4-7,2005, Asilomar, CA, USA.New York,NY,USA: ACM, 2003:13p.
[6] LUCKHAM D C, FRASCA B.Complex Event Processing in Distributed Systems[R]. CSL-TR-98-754.Stanford,CA,USA:Stanford University, 1998.
[7] BRENNA L,DEMERS A, GEHRKE J,et al. Cayuga: A High-Performance Event Processing engine[C]//Proceedings of the ACM SIGMOD International Conference on Management of Data(SIGMOD'07), Jun 11-14,2007,Beijing,China.New York,NY,USA: ACM,2007:1100-1102.
[8] Oracle cep[EB/OL].http://www.oracle.com/technologies/soa/complexevent-processing.html.
[9] 高洪.基于P2P网络的分布式消息队列[J]. 程序员, 2012(6):102-106.

作者简介

陆平，东南大学毕业；中国计算机协会CCF会员、服务计算专委；现任中兴通讯业务研究院院长，负责业务软件、多媒体终端、信令检测、ICT等产品的研发以及互联网、云计算、家庭网络等新业务的研究；曾主持多项国家重点课题研究；发表论文10篇。

钱煜明，东南大学毕业；中兴通讯业务研究院总工程师，负责大数据处理、云计算、移动互联网等方向系统架构及新技术研究；江苏省双创人才，主持多项国家重点课题研究；发表论文16篇。

朱科支，东南大学毕业；现任中兴通讯业务研究院产品经理，负责移动互联网及大数据处理相关产品研发及管理，对于大数据处理、搜索引擎、并行计算、移动终端管理等方面有深入研究。

DOI:10.3969/j.issn.1009-6868.2013.04.013
网络出版地址:http://www.cnki.net/kcms/detail/34.1228.TN.20130625.1440.002.html

近场通信技术 1

孙成丹/SUN Chengdan，彭木根/PENG Mugen
(北京邮电大学信息与通信工程学院,北京 100876)

[编者按]近场通信技术近年来逐渐受到人们的关注,相关的技术标准和协议规范也日臻完善。讲座将分3期对该技术进行介绍:第1期讲述近场通信的背景及概况,概述性介绍近场通信技术的技术架构;第2期对近场通信的具体技术规范做详细介绍,包括数字协议规范、相关动作规范、逻辑链路控制协议、标签类型及数据交换格式;第3期介绍近场通信的安全技术、设备的连接切换规范和业务应用。

中图分类号:TP393.03 文献标志码:A 文章编号:1009-6868 (2013) 04-0063-04

1 近场通信背景及概述

在2013年2月闭幕的巴塞罗那世界移动通信大会上,近场通信(NFC)技术在全球移动通信协会的强力推介下登场,成为业界新热点。各大厂商也加紧了对于支持NFC技术产品的研发生产。近场通信技术又称近距离无线通信,鉴于各种移动互联网应用的广泛开展和未来发展的广阔空间,它已是信息时代的又一新宠,也是各大厂商和服务商争夺的下一块领地。

1.1 NFC背景

NFC技术是于2004年4月由飞利浦公司发起,是一项由飞利浦、诺基亚、索尼等厂商联合主推的近距离无线技术。多家公司和大学共成立了泛欧联盟,旨在推动NFC开放式架构的开发和其在手机中的应用。NFC由射频识别(RFID)及互联互通技术整合演变而来,保持对RFID的兼容性。通过在单一芯片上结合感应式读卡器、感应式卡片和点对点的功能,具备NFC功能的设备能在短距离内与兼容设备进行识别和数据交换。这项技术最初只是RFID技术和网络技术的简单合并,现在已经演变成一种短距离无线通信技术,近年来逐年受到关注。很明显,近场通信利用的是无线电波的临近电磁场,根据电磁理论,近磁场的信号传播过程中强度会以大约$1/d^6$的速率下降(d表示通信距离),如此大的衰减使近场通信成为名副其实的短程通信技术。相比之下,在无线电波的远场中,信号强度以$1/d^2$的速率下降。

近场通信技术在ISO 18092、ECMA 340和ETSI TS 102 190框架下推动标准化,同时也兼容应用广泛的ISO 14443 A/B以及Felica标准非接触式智能卡的基础架构。

作为一种近距离的高频无线通信技术,近场通信的可用距离约为10 cm,可以实现电子身份识别或者数据传输,其应用范围已由电子支付扩展至旅行、交通、购物等方面。NFC技术的短距离交互很大程度简化了设备互联过程中整个认证识别过程,使得电子设备间互相访问更直接、简答、安全并更清楚。

NFC技术结合了非接触式感应以及无线连接的相关技术,并作用于13.56 MHz频带,同时支持106 kbit/s、212 kbit/s或者424 kbit/s等传输速度,将来最高支持速率可提高至1 Mb/s左右,为设备间不同的应用场景提供了灵活的选择能力。与其他短距离无线通信技术相比,NFC技术更安全,反应时间更短。并且,由于近场通信技术与现有非接触智能卡技术相兼容,目前已经得到越来越多厂商的支持并成为正式标准,这些都为NFC技术大范围的应用提供了可能。NFC技术提供各种设备间轻松、安全、迅速而自动的通信,例如借助NFC技术,人们可以在不同的设备间交换照片、音乐、视频剪辑等信息。

其实,近场通信并非新生事物,但直到近年来才逐渐受到关注。在NFC技术发展过程中有几个重要的历程值得一提:1983年查尔斯·沃尔顿获得第一个RFID相关专利;2004年诺基亚、飞利浦和索尼联合组建了近场通信论坛;2006年NFC标签的初

收稿日期:2013-06-07
网络出版时间:2013-06-25
基金项目:国家科技重大专项课题(2012ZX03001037-004, 2012ZX03001028-04, 2012ZX03001031-004);北京市科技新星合作项目(xxhz201201)

步规范；2006年规范“SmartPoster”的记载；2006年诺基亚6131成为首个NFC功能的手机；2010年三星Nexus S成为首款可以支持NFC功能的Android手机。

作为一种无线技术，NFC同样面临安全问题。但是NFC技术本身的特点——非常小的通信范围，有效隔绝了黑客的入侵，用户完全可以放心地在这样的近距离中进行通信。但是为了提供安全可靠的通信，近场通信技术也包含了完整的安全技术。

1.2 NFC的3种工作模式

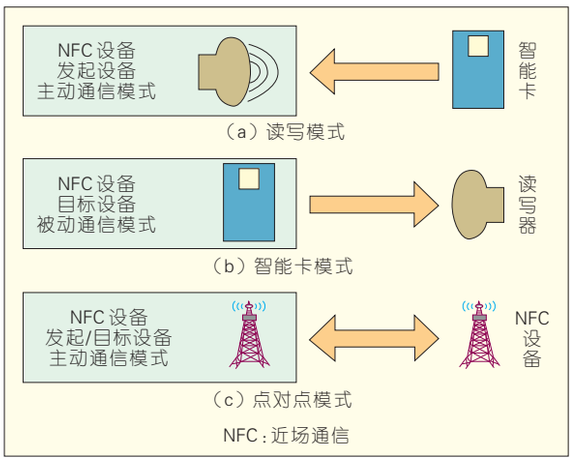
近场通信技术支持3种不同的工作模式：卡模式、点对点模式和读卡器模式，如图1所示。

在卡模式下，NFC设备相当于一张采用RFID技术的IC卡，完全可以应用于现在IC卡（包括信用卡）的使用场合，如公交卡、商场消费卡、车票，门禁管制、门票等等。这种方式下的一个明显优点是卡片通过非接触读卡器的RF域来供电，即便是在寄主设备（如手机、移动终端）没电的情况下也可以保证数据的传输工作。

在点对点模式下，NFC技术和红外线技术一样，可用于数据交换，只是采用NFC技术的设备传输距离较短，传输创建速度较快，传输数据的速度也较快。相比于红外设备，采用NFC技术的设备功耗较低。将两个具备NFC功能的设备连接后，即可实现数据在设备间的点对点传输，可完成下载音乐、交换图片或者同步设备地址薄等功能。因此通过近场通信技术，多个设备（如数位相机、PDA、计算机和手机等）之间可以交换资料或者互相提供服务。

在读卡器模式下，NFC设备可以作为非接触读卡器使用，从海报或者展览信息电子标签上读取相关信息。

需要进行数据交互时，NFC设备可以工作于主动模式或被动模式下。在被动模式下，发起NFC通信的设备，也称为NFC发起设备（主设备），在整个通信过程中提供射频场。它可以从106 kbit/s、212 kbit/s或424 kbit/s中选择一种传输速度，将数据发送到另一台设备。另一台NFC设备作为目标设备(从设备)，不必主动产生射频场，仅需要通过负载调制技术，以相同的速度将数据传回发起设备。此通信机制与基于ISO14443A、FeliCa的非接触式智能卡兼容。因此，NFC发起设备在被动模式下，可以用相同的连接和初始化过程检测非接触式智能卡或NFC目标设备，并与之建立联系。在主动模式下，通信双方收发器加电后，任何一方可以采用“发送前侦听”协议来发起一个半双工发送。在一个以上NFC设备试图访问一个阅读器时，这个功能可以防止冲突。



▲图1 3种工作模式

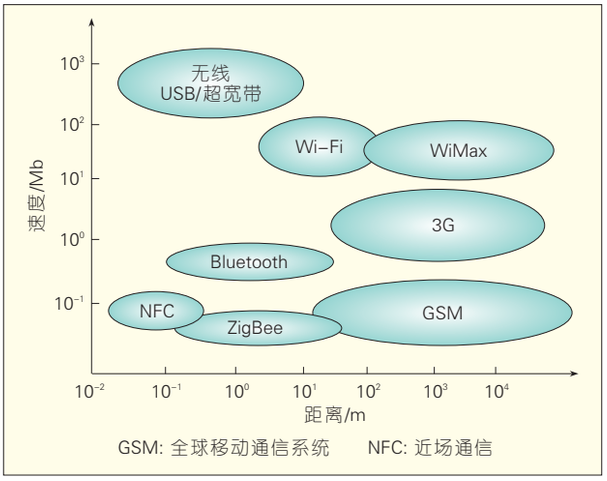
在主动模式下，每台设备要向另一台设备发送数据时，都必须产生自己的射频场。发起设备和目标设备都要产生自己的射频场，以便进行通信。在被动模式下，像RFID标签一样，目标是一个被动设备。标签从发起者传输的磁场获得能量，然后通过负载调制技术将数据传送给发起者。

需要注意的是，移动设备主要工作于被动模式下，从而能够大幅降低功耗，延长电池寿命。在一个应用会话过程中，NFC设备可以在发起设备和目标设备之间切换自己的角色。利用这项功能，电池电量较低的设备可以要求以被动模式充当目标设备，而不是发起设备。

1.3 与其他无线通信技术的比较

目前，无线通信市场多种技术并存，尤其是近距离通信领域，已经存在多种近距离无线通信技术，比如蓝牙技术、红外线技术、RFID技术等，这些技术的并存为用户提供了丰富多样的业务，并且每一种技术都有自己的应用场景和优势。近场通信技术的出现，丰富了近距离无线通信技术的种类，完善了近距离无线通信的应用场景和范围，也为用户提供了更大的选择灵活性。与现存的诸多近距离无线通信技术相比，NFC技术具有明显的优势。图2展示了无线通信市场中各技术适用场景[1]。

与RFID相比，近场通信技术中的信息也是通过无线频率的电磁感应耦合方式传递，利用了负载调制的功能。但两者之间还是存在很大的区别。首先，与RFID技术相比，NFC的传输距离更短，可以提供轻松、安全、迅速的无线连接。已知的RFID的传输范围可以达到几米、甚至几十米，近场通信技术由于其独特的技术优势，对信号进行了有效衰减，从而有效地降低了电磁波的传输距离，因此NFC具有比RFID技术更近的传输距离、更高的带宽、更低的能耗等特点。其次，近场通信技术天生的优势是与现有非接触智能卡技术兼容，目前已经成为越来越多主要厂商支持的正式标准，因此具有更广阔的应用前景和使用范围。同时，NFC技术是一种近距离连接协议，提供设备间轻松、迅速、安全而自动的通信。与其他无线连接方式相比，近场通信是一种近距离的私密通信方式。最后，近场通信与RFID的应用领域不同，NFC



▲图2 无线通信技术应用情况

技术主要应用于门禁、公交、手机支付、交通、旅行、购物等领域，RFID技术则在生产、物流、跟踪、资产管理等领域内发挥着巨大的作用。

此外，与红外和蓝牙传输方式相比，近场通信技术也表现出自己独特的优势。与红外技术相比，NFC技术提供一种面向消费者的、更近距离的交易机制，比红外传输方式更快、更可靠、更简单。与蓝牙传输技术相比，一方面，近场通信技术面向近距离交易，适用于交换财务信息或敏感的个人信息等重要私密数据；另一方面，蓝牙技术能够弥补NFC技术通信距离不足的缺点，可以应用于较长距离的数据通信。因此，NFC技术和蓝牙技术可以相互补充、共同存在。事实上，快捷轻型的NFC协议可以用于引导两台设备之间的蓝牙配对过程，促进蓝牙的使用。表1中直观地表示了近场通信技术、红外技术和蓝牙技术在几个技术性能指标上的差异。

正是由于近场通信技术具有独特的技术优势，以及其对多种标准规范的有效支持和兼容，加上NFC具有成本低廉、方便易用和更富直观性等特点，这让它在某些领域显得更具潜力。NFC通过一个芯片、一根天线和一些软件的组合，能够实现各种设备在几厘米范围内的通信，并且费用低廉。近几年随着智能手机的普及，NFC技术逐渐走入寻常百姓家，众多厂商纷纷在自己的终端设备中加入了NFC功能，抢占NFC市场先机。可以预言：如果NFC技术能得到普及，它将在很大程度上改变人们使用许多电子设备的方式，甚至改变使用信用卡、钥匙和现金的方式。我们有理由相信：近场通信技术将在移动互联网时代大放异彩。

2 近场通信技术架构

近场通信技术支持3种不同的工作模式，每种工作模式具有相似的技术架构，但是具体的工作模式又体现出各自的差别。我们将概述性地介绍近场通信的技术架构。

按照从下至上的顺序，近场通信技术的总体技术架构包括以下几个部分：模拟协议规范、数字协议规范、NFC相关动作规范、逻辑链路控制协议、NFC标签技术规范、NFC数据交换格式、记录类型定义规范等。每一种技术规范都完成特定功能，并且针对具体的业务应用和工作模式灵活选择适当的协议规范实现。具体架构如图3所示[1]。

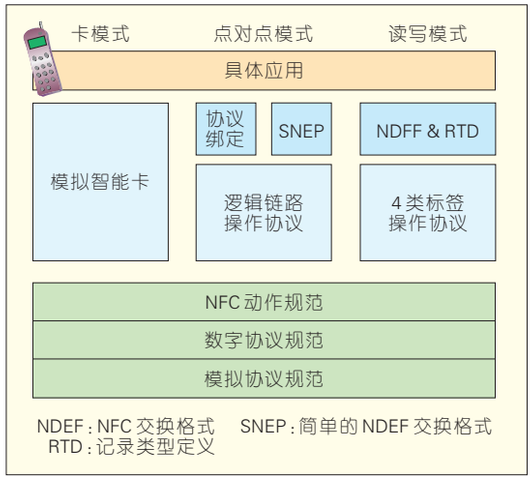
模拟协议规范的作用主要是定义了具备NFC功能的设备的无线射频特性，如射频域的形状和强度。该规范主要用来决定NFC设备的可操作范围。根据规定，近场通信技术的射频磁场的载波频率为13.56 MHz，未经调制的射频磁场磁场强度最小值为H_{min}=1.0 A/m rms，未经调制的射频磁场强度最大值H_{max}=7.5 A/m rms。在通信的过程中需要对磁场进行调制。被动通信模式下，初始方应产生一个射频磁场来给目标方供应能量，目标方应该能够在H_{min}和H_{max}间连续工作，在实际使用过程中，当目标方在初始方的工作区域中时，初始方在其工作区域中的磁场强度应不小于H_{min}。在主动通信模式下，初始方和目标方都是用自身产生的射频磁场(H_{min}~H_{max})进行通信。在实际使用过程中，当目标方和初始方在对方的工作区域中时，初始方和目标方应保证在自身工作区域中的磁场强度不小于H_{min}。当进行外部磁场检测时，如果外部磁场在频率为13.56 MHz处的场强高于$H_{Threshhold}$，NFC设备应能检测出该外部磁场的存在。外部射频磁场阈值$H_{Threshhold}$=0.1875A/m。

数字协议规范主要定义了用于完成通信的构件，是实现ISO/IEC 18092和ISO/IEC 14443标准中数字技术的规范。涉及到4种不同角色(初始方、目标方、读写器、卡模拟器)下的NFC设备的数字接口和半双工传输协议。主要包括调制机制、比特级编码、比特速率、帧格式、相关协议和

▼表1 3种技术比较

	NFC	蓝牙	红外
网络类型	点对点	单点对多点	点对点
最大使用距离/m	0.1	10	1
速度	106，212，424规划速率可达868 kbit/s	721 kbit/s或更高	115 kbit/s
建立连接时间/s	<0.1	6	0.5
安全性	具备，由安全IC卡硬件实现	具备，软件实现	不具备，使用IFRM时除外
通信模式	主动-主动 主动-被动	主动-主动	主动-主动
成本	低	中	低

NFC: 近场通信



▲图3 技术架构

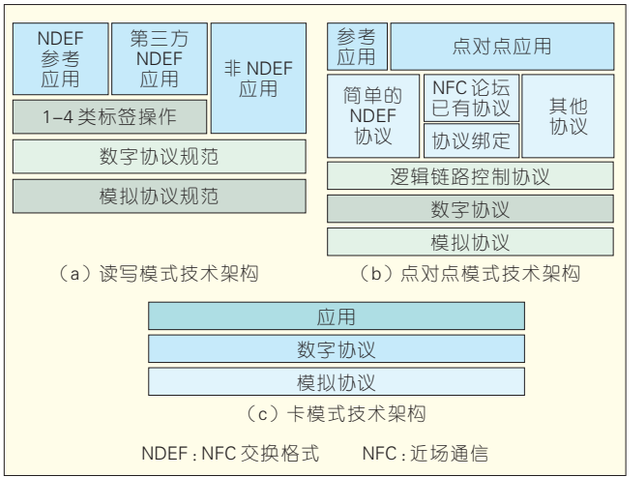


图4▶ 3种模式技术架构

命令集。

NFC相关动作规范以满足数字协议规范的构件为基础，定义了互动方式下建立通信的一系列动作。如轮询周期、何时执行进行冲突检测等动作。规范中定义的一些动作可以原样使用，或者通过适当修改来定义其他的方式来建立通信，这些变种方式可以适用于原用例或者适用于不透光的用例。

逻辑链路控制协议(LLCP)描述了NFC套件逻辑链路控制层(LLC)的功能、特征和协议。逻辑链路控制层构成了OSI模型数据链路层的上半层，与下半层的媒体接入控制层(MAC)互补。LLCP层技术规范通过一系列映射可以支持MAC层。LLCP协议到外部MAC协议的每一种映射都指定了相应的绑定需求。LLCP主要特征包括链路激活、监测、去活，异步均衡通信，高层协议复用，无连接传输，面向连接的传输等。LLCP不支持同步传输、多播与广播、数据的安全传输、服务用户接口等功能。

NFC标签技术规范定义了4种NFC的标签类型，以支持设备的读卡器工作模式。

数据交换格式(NDEF)规范定义了NFC应用中的信息编码格式。该规范支持NDEF信息的复用和分块。

记录类型定义规范如何在NDEF信息中构造记录，并指出记录可以互相包含。每一种记录都包含一个类型指示，表明其包含的内容。

NFC设备有3种工作模式，对不同的工作模式，在协议使用与应用定义方面有明显的差异。

工作于读写模式下的设备支持的应用可分为3类：NDEF参考应用、第三方NDEF应用和非NDEF应用。NDEF参考应用是指NDEF论坛预定义的一些具有参考价值的应用，如链接切换、智能海报等；第三方NDEF应用是指使用NDEF技术基于标签的专利性私有应用；非NDEF应用是指需要与非接触式卡片交互的专利性私有应用[1–2]。读写模式下的技术架构如图4(a)所示。

点对点模式下的技术架构如图4(b)所示。逻辑链路控制协议负责链路的激活、管理、去激活，该协议支持异步平衡模式和协议复用技术，同时支持无连接传输和面向连接的传输情况；协议绑定模块为NFCC论坛定义的协议规范提供标准的绑定(即端口号)，增强不同协议之间的互操作性；论坛已有协议部分是指那些论坛已定义了的与LLCP相互绑定的协议，如IP、对象交换协议(OBEX)；其他协议是指那些论坛为指定的可以运行于LLCP协议层之上的部分协议；参考应用指论坛定义的可以运行于NDEF协议上的参考性应用；点对点应用可能包括从相机打印照片、交换商务名片，以及第三方NDEF应用等等。

卡模式下的NFC设备架构比较简单。其应用主要包括一些专利性的非接触式卡片应用，如基于ISO14443 A/B或FeliCa标准的付账、购票应用等。具体技术架构如图4(c)所示。 (待续)

参考文献
[1] NFC Forum. NFC digital protocol technical specification 1.0[S].2010.
[2] NFC Forum. NFC Data Exchange Format (NDEF) technical specification 1.0[S].2006.

作者简介

孙成丹，北京邮电大学在读硕士研究生，目前主要研究方向为无线网络信息理论和关键技术。

彭木根，北京邮电大学教授、博士生导师，IEEE高级会员；主要从事为时分双工无线网络信息理论、协同网络编码、无线网络自组织技术、TDD高能效无线传输和组网技术、TD-SCDMA及增强演进系统的传输和组网增强技术的研发工作；荣获高等学校科学研究优秀成果奖(科学技术)技术发明奖一等奖、中国通信学会科技进步奖二等奖和北京青年优秀科技论文奖二等奖，国际学术会议最佳论文奖3次；在国际著名学术期刊发表SCI学术论文约50篇，获得授权发明专利30余项，提交标准技术文稿30余篇，出版学术专著和译著10余部。